

“UNIVERSIDAD NACIONAL DE PIURA”

FACULTAD DE CIENCIAS

**ESCUELA PROFESIONAL DE INGENIERÍA
ELECTRÓNICA Y TELECOMUNICACIONES**



**“DISEÑO Y CONSTRUCCION DE UN PROTOTIPO PARA LA
SUPERVISION DE VARIABLES EN LA INDUSTRIA DE
ALIMENTOS USANDO LA TELEMATICA-CLOUD COMPUTING
Y S.O. ANDROID”.**

PRESENTADA POR:

**BACH. IRVING SAUL NUÑEZ MACHADO
BACH. JUSTO EDUARDO TABOADA ADRIANZEN**

**TESIS PARA OBTAR EL TÍTULO PROFESIONAL DE
INGENIERO ELECTRÓNICO Y TELECOMUNICACIONES**

**PIURA – PERÚ
2015**

“UNIVERSIDAD NACIONAL DE PIURA”

FACULTAD DE CIENCIAS

**ESCUELA PROFESIONAL DE INGENIERÍA
ELECTRÓNICA Y TELECOMUNICACIONES**



**“DISEÑO Y CONSTRUCCION DE UN PROTOTIPO PARA LA
SUPERVISION DE VARIABLES EN LA INDUSTRIA DE
ALIMENTOS USANDO LA TELEMATICA-CLOUD COMPUTING Y
S.O. ANDROID”.**

PRESENTADA POR:

**BACH. IRVING SAUL NUÑEZ MACHADO
BACH. JUSTO EDUARDO TABOADA ADRIANZEN**

**TESIS PARA OBTAR EL TÍTULO PROFESIONAL DE INGENIERO
ELECTRÓNICO Y TELECOMUNICACIONES**

**PIURA – PERÚ
2015**

TESIS PRESENTADA COMO REQUISITO PARA OPTAR EL TÍTULO DE
INGENIERO ELECTRÓNICO Y TELECOMUNICACIONES

ASESOR:



ING. FRANKLIN BARRA ZAPATA

TESISTAS:



BACH. IRVING SAUL NUNEZ MACHADO

BACH. JUSTO EDUARDO TABOADA ADRIANZEN

JURADO EVALUADOR:

PRESIDENTE:



ING. EDWIN ARNALDO OCAS INFANTE

SECRETARIO:



ING. EDUARDO OMAR AVILA REGALADO

VOCAL:



ING. JUAN MANUEL JACINTO SANDOVAL

DEDICATORIA

A mis padres, hermanos por su apoyo
incondicional durante mi formación profesional

AGRADECIMIENTO

Esta tesis no hubiera sido posible sin el impulso de todas aquellas personas que me han ayudado personalmente en el desarrollo del trabajo de esta índole, comenzando por el Ing. Franklin Barra Zapata a quien agradezco su apoyo en la elaboración del presente proyecto.

Un sincero agradecimiento al jefe del área de trabajo, por darme permiso siempre que lo requería.

Un agradecimiento muy especial por la paciencia y el ánimo recibidos de mi familia y amigos.

A todos ellos, muchas gracias.

INDICE GENERAL

CAPITULO I	8
PLANTEAMIENTO DEL PROBLEMA	8
1.1.INTRODUCCION	8
1.2.ANTECEDENTES	9
1.3.PLANTEAMIENTO DEL PROBLEMA	10
1.3.1. DELIMITACION	10
1.3.2.APORTES DEL PROYECTO	10
1.3.3.Formulación del Problema	11
1.3.4.HIPOTESIS	11
1.3.5.OBJETIVOS	11
CAPITULO II	12
MARCO TEORICO	12
2.1. Cloud Computing: (Computación en la Nube)	12
2.1.1 Definición de Computación en la Nube	12
2.1.2 Características	13
2.1.3. Modelos de Implementación	13
2.1.4. Esquemas de Funcionamiento	14
2.1.5. Financiamiento	14
2.1.6. Plataformas para el Desarrollo de Software en la Nube	14
2.1.6.1 Google App Engine	15
2.1.6.2. Windows Azure	16
2.1.6.3. Amazon Web Services	16
2.1.6.4. Engine Yard	17
2.1.6.5 Heroku	18
2.1.7. Arquitecturas para el Desarrollo de Software en la Nube	18
2.1.7.1. Capas	18
2.1.7.2. Arquitecturas	20
2.1.8. Futuro de la computación en la nube	22
2.1.8.1. Tendencias en Torno a la Computación en la Nube.	23
2.1.9. Pruebas de concepto	25
2.1.9.1. Aplicaciones interesantes en la Nube [14]	25
2.1.9.2 Pruebas de concepto en la Nube	25
2.2.DESCRIPCIÓN GENERAL DE LA TECNOLOGÍA INALÁMBRICA BLUETOOTH	26
2.2.1.CARACTERÍSTICAS DE BLUETOOTH	26
2.2.2.CANALES DE TRANSMISIÓN BLUETOOTH	26
2.2.3.DISPOSITIVO BLUETOOTH HC06	27
2.2.4. Características del Módulo HC06	28
2.2.5. Conectando el Modulo Bluetooth HC-06 con Microcontrolador	29
2.2.6.Obtener MAC del Modulo Bluetooth HC06	31
2.3.DESCRIPCION DE LOS MICROCONTROLADORES	32

2.3.1. MICROCONTROLADORES PIC	32
2.3.2.MICROCONTROLADOR PIC16F877A	33
2.3.3.MICROCONTROLADOR PIC16F628A	36
2.4.COMPILADORES	38
2.4.1.COMPILADOR PICBASIC PROCOMPILER	38
2.5.SISTEMA OPERATIVO ANDROID	39
2.6.PROGRAMACION EN APPINVENTOR	42
2.6.1.HISTORIA	42
2.6.2.CARACTERISTICAS	43
2.7. Sensores de gas	45
2.7.1. Módulo Sensor de humo y gases inflamables MQ-2:	45
2.8. Componente TinyWebDB de AppInventor	47
2.9 La Industria Alimentaria	49
2.9.1. Los Procesos de la Cadena Alimentaria	49
2.9.2. Sectores de la industria	51
2.9.3. VARIABLES UTILIZADAS EN LA INDUSTRIA ALIMENTARIA	52
CAPITULO III	54
DISEÑO HARDWARE Y SOFTWARE DEL SISTEMA	54
3.1. Diagrama de Bloques del Sistema.	54
3.2. Diagrama Electrónico de la Tarjeta para sistema	55
3.3. Programación del Microcontrolador	58
3.4. Software para dispositivo móvil	60
3.4.1. Especificaciones de la aplicación para el dispositivo movil	60
3.4.2. Desarrollo de aplicativo en Appinventor	62
3.4.3. Componentes visibles pantalla principal	63
3.4.4. Componentes No Visibles pantalla principal	65
3.4.5. Componentes visibles pantalla de descarga de datos	66
3.4.6. Componentes No Visibles pantalla de descarga de datos	67
3.4.7. Descripción del Programa de pantalla principal	68
3.4.7.1. Variables	68
3.4.7.2.BOTONES	69
3.4.7.3.DESLIZADOR	71
3.4.7.4.TEMPORIZADOR	72
3.4.7.5.NOTIFICACIÓN	75
3.4.7.6.Descargar datos de Servidores de Google	75
CAPITULO IV	79
COSTOS DEL PROYECTO	79
4.1. COSTOS DEL PROYECTO	79
Conclusiones	80
Bibliografía	81
Anexos	82

CAPITULO I

PLANTEAMIENTO DEL PROBLEMA

1.1.INTRODUCCION

Es imposible hablar en estos momentos de la ciencia y del desarrollo en cualquier esfera tanto social como económica de un país, sin referirse a la electrónica, las comunicaciones, y la informática y los servicios que estas brindan para hacer más fácil y eficiente el trabajo humano.

La Telemática es una disciplina científica y tecnológica, originada por la convergencia entre las tecnologías de las Telecomunicaciones y de la Informática. El desarrollo de esta rama de la ciencia crece a ritmos acelerados.

El cloud computing consiste en la posibilidad de ofrecer servicios a través de Internet. La computación en nube es una tecnología nueva que busca tener todos nuestros archivos e información en Internet y sin depender de poseer la capacidad suficiente para almacenar información. El cloud computing explica las nuevas posibilidades de forma de negocio actual, ofreciendo servicios a través de Internet, conocidos como e-business (negocios por Internet).

Sin embargo, los servicios en la nube están creciendo a pasos agigantados, principalmente por los beneficios que obtienen las compañías al adoptar una solución empresarial basada en la nube:

- **Demanda tipo autoservicio:** La empresa puede unilateralmente utilizar las capacidades computacionales de la forma que requiera, tales como tiempo de servidor o almacenamiento en red, sin necesidad de interactuar con cada proveedor de estos servicios.
- **Acceso a la red en forma móvil:** Las capacidades computacionales están disponibles en la nube, lo que permite acceder a ellas desde diferentes dispositivos,

ya sean Smartphone, computadores portátiles, tablets, etc., lo cual es muy relevante en la era de la movilidad.

- Pago por uso: Las capacidades computacionales se asignan a la empresa mediante un pago-por-servicio o modelo de facturación basado en promover la optimización del uso de recursos. Por ejemplo, la medición basada en la capacidad de almacenamiento, ancho de banda, o por el número de cuentas de usuario activas por mes.

En general, las soluciones en la nube permiten adoptar aplicaciones en un menor tiempo, entregan una gran facilidad de uso y son muy intuitivas, lo cual aporta un valor significativo a las empresas en el mundo

1.2.ANTECEDENTES

Según la Lic. Beatriz L. García Delgado en su artículo. “La telemática en la Industria Porcina” :El uso de la telemática aplicada a la industria porcina no se contenta con la humanización del trabajo, su aplicación ha ido mucho más allá, al desarrollo de novedosos modelos Computacionales que unido al empleo de las más actuales técnicas de comunicación han permitido crear sistemas capaces de almacenar y procesar grandes volúmenes de información generados en diferentes puntos geográficos ya grandes velocidades, los cuales en muchos casos posibilitan la predicción de la producción y del comportamiento animal, no solo dentro la esfera de la producción porcina (manejo, nutrición, genética, reproducción, veterinaria y otros) sino a lo largo de todas los eslabones que componen la cadena productiva del cerdo. Resulta importante para todos los que, de una forma u otra, tienen que ver con el desarrollo de la actividad investigativa y productiva de esta rama, el conocimiento de lo que en materia de informática, comunicaciones y electrónica se ha logrado hasta la fecha. En este artículo nos permite visualizar que la telemática y sus diferentes ramas se aplican en los procesos Industriales actualmente.

En el Artículo “**Cloud Computing la tercera ola de las tecnologías de la Información**”, publicado por Fundación por la Innovación Bankinter, Indica que la crisis económica ha despertado el interés de las empresas y los Gobiernos por la

nube como un medio para reducir costes, pero éste no es su único atractivo, puesto que también ofrece un amplio abanico de posibilidades, algo de lo que los usuarios se están dando cuenta poco a poco la nube resulta especialmente atractiva para las pymes. [7]

Primero, ahora que estas empresas no tienen acceso fácil al capital, la reducción de inversiones iniciales que ofrece la nube es un medio para no perder competitividad. [7]

Segundo, la nube reduce el time-to-market y gracias a este modelo nuevos servicios pueden estar listos en cuestión de horas con un riesgo limitado Tercero, acceden a economías de escala a través de los proveedores. Cuarto, acceden a Sistemas de Seguridad mucho más sofisticados. Y, por último, se benefician de un soporte al usuario muy especializado [7].

1.3. PLANTEAMIENTO DEL PROBLEMA

1.3.1. DELIMITACION

Este proyecto pretende desarrollar e implementar un sistema de supervisión de variables físicas como son Nivel, temperatura, en la Industria Alimentaria y luego estos valores ir almacenándolos en la red de internet en un servidor externo como pueden ser los servidores de Google. Y lo cual se podrá acceder en cualquier momento y de cualquier lugar. A este tipo de accesos y almacenamiento lo podremos llamar almacenar y gestionar en la Nube (cloud computing), para lo cual se desarrollara aplicativo a celulares con sistema Operativo Android.

1.3.2. APORTES DEL PROYECTO

- Incentivar la investigación y el uso de nuevas tecnologías y sistemas de comunicación en aplicaciones de diseño de equipos y sistemas.
- Con el uso del Cloud Computing se puede supervisar y monitorear las variables Físicas de un Proceso Industrial.
- Desarrollo de aplicativos en Sistema Operativo Android para la Industria de Procesos de Alimentos.

1.3.3 Formulación del Problema

¿Es posible Diseñar y construir un prototipo para supervisión de variables en la Industria de Alimentos usando la Telemática- Cloud Computing y Sistema Operativo Android, para mejorar los procesos y disponer de los datos en cualquier instante y de cualquier lugar accediendo a la red de Internet?

1.3.4.HIPOTESIS

Las tecnologías existentes si nos permite Diseñar y construir un Sistema de Supervisión de variables para la industria de alimentos usando la Telemática - Cloud Computing (uso de servidores externos) y sistema Operativo Android como medio de acceso al sistema.

1.3.5.OBJETIVOS

Objetivo General

Diseñar y construir un Sistema de Supervisión de procesos industriales usando la Telemática - Cloud Computing y S: O; Android

Objetivos Específicos:

1. Determinar cuáles son las variables Físicas más importantes en la Industria de Procesamiento de Alimentos
2. Determinar cuáles son los Procesos más usados en el Procesamiento de alimentos
3. Construir un sistema donde se pueda supervisar las variables más importantes
4. Construir o habilitar una tarjeta electrónica para la medición de variables
5. Diseñar e implementar un sistema de comunicación entre tarjeta electrónica y sistema de computo
6. Implementación del Cloud Computing con Sistema Operativo Android.

CAPITULO II

MARCO TEORICO

2.1. Cloud Computing: (Computación en la Nube)

El modelo de computación en la nube apuesta a una nueva era de la computación, rompiendo con las actuales estructuras de mercado y arquitectura de productos software. Cloud Computing parece surgir con el propósito de solucionar las continuas dificultades que se presentan en los proyectos y productos de la industria del software como los altos costos de adquisición en recursos de TI, el mantenimiento y la estabilidad de las plataformas tecnológicas entre otros.

2.1.1 Definición de Computación en la Nube

El término computación en la nube ha sido tomado posiblemente a la continua simbología que se utiliza para referirse al término Internet. Con lo que realmente estaríamos hablando de “Computación en Internet”.

Computación en la nube es un modelo para permitir acceso por demanda a la red, a un pool compartido de recursos informáticos configurables (por ejemplo redes, servidores, almacenamiento, aplicaciones y servicios) que puede ser rápidamente aprovisionado y liberado con muy poco esfuerzo de gestión.

El modelo de computación en la nube promueve disponibilidad y está compuesto por cinco características esenciales (autoservicio en demanda, extenso acceso a la red, puesta común de recursos, rápida elasticidad, servicio medido), tres modelos de servicio (SaaS, PaaS, IaaS) y cuatro modelos de despliegue (privado, community, publico, híbrido) [1].

La computación en la nube se ha posicionado en el mercado como una tecnología o para algunos como una opción más que considerar en la búsqueda de soluciones personales y empresariales que ofrecen servicios a través de internet. Este tipo de soluciones se pueden encontrar tanto a nivel entretenimiento como empresarial.

Entre las principales características de la computación en la nube encontramos el autoservicio en demanda. Las aplicaciones en la nube permiten, como en un autoservicio, servirse de los productos según las necesidades del usuario y también de las capacidades de adquisición. En esta modalidad es posible acceder tanto a aplicaciones gratuitas como a aquellas que no lo son, y en muchos casos las primeras ofrecen soporte a nivel empresarial en modalidades de pago.

2.1.2 Características

Entre algunas de las características destacadas en el modelo de computación en la nube encontramos las siguientes:

Servicio Bajo Demanda. Mediante este modelo los clientes pueden acceder a servicios como almacenamiento y servidores de red, el tiempo de uso y el volumen de tráfico entre otros. Esta característica permite un modelo de autoservicio que se acomoda, como un guante, a las necesidades específicas del cliente, que pueden ser variables en el tiempo. De este modo, un cliente que requiere alto tráfico en la red por algunos periodos de tiempo como podría ser el caso de los servicios requeridos para encuestas, votaciones o censos, no tendría que invertir en un servicio permanente que requiere de altos costos.

Elasticidad y Escalabilidad. Cloud computing permite una entrega de servicios de manera rápida y elástica, incluso en algunos casos puede ser de manera automática. El modelo en la nube proporciona alta escalabilidad, permitiendo a los clientes la utilización de una pequeña parte de la aplicación en un momento dado y luego acceder a la totalidad de esta.

Supervisión del Servicio. En el modelo de computación en la nube el uso de recursos puede ser monitoreado, controlado y notificado, garantizando la transparencia, tanto para el proveedor como el cliente.

2.1.3 Modelos de implementación

Existen básicamente tres modelos de implementación de computación en la nube, estos modelos están orientados al servicio y poseen las características mencionadas en el apartado anterior. Los modelos difieren en el tipo de aplicación y el nivel de acceso a los servicios requeridos por los clientes

Nube Privada. En este modelo la infraestructura de la nube es operada solamente por una organización. Puede ser administrada por la organización o por un tercero.

Nube Pública. La infraestructura de la nube en este modelo está disponible para el público en general o para un grupo de industrias y su propietario es una organización que provee los servicios de computación en la nube.

Nube Comunitaria. En este caso, la infraestructura de la nube es compartida por varias organizaciones y soporta una comunidad específica que tiene objetivos en común.

Nube Híbrida. La infraestructura de la nube es una composición de dos o más nubes (privada, comunitaria o pública), que siendo entidades únicas, permanecen unidas por estándares o tecnologías, permitiéndoles portabilidad de las aplicaciones y de los datos. [1]

2.1.4. Esquemas de Funcionamiento

La computación en la nube se basa en una arquitectura multiusuario, que consiste en una única aplicación la cual es compartida por todos los clientes, lo contrario de las aplicaciones de software tradicional que consisten en copias distintas del producto para cada usuario. En la arquitectura multiusuario todos los clientes comparten la infraestructura y los servicios, y cada negocio puede personalizar su propio entorno. (Aplicaciones elásticas) [2]

2.1.5. Financiamiento

En este modelo existe diversidad de formas de pago. La principal característica de los modelos de computación en la nube es que el cliente no tiene que adquirir ningún producto que requerirá mantenimiento y actualización sino que obtendrá una suscripción a los servicios específicos adquiridos. Algunos proveedores de computación en la nube ofrecen servicios gratuitos como “Google Docs” y sus medios de financiamiento provienen de diversas formas, entre ellas los planes empresariales, las pautas comerciales, entre otros.

2.1.6. Plataformas para el Desarrollo de Software en la Nube

Existen muchas empresas dedicadas a prestar servicios de plataformas de Cloud Computing. A continuación se enumeran algunas de las más populares de ellas:

2.1.6.1 Google App Engine

Es una plataforma para el desarrollo y hosting de aplicaciones web en los Data centers de Google que virtualiza soluciones a través de múltiples servidores. Los lenguajes que soporta actualmente son Python y Java, y por extensión, otros lenguajes que pueden funcionar sobre la máquina virtual de Java.

Google App Engine tiene algunas restricciones como:

- Aplicaciones Java solo pueden usar algunas clases del Java Runtime Environment que se encuentran en una "lista blanca".
- Aplicaciones Java no pueden crear nuevos hilos.
- SSL es solo soportado a través de dominios *.appspot.com.
- Solo se soportan módulos Python que sean completamente en este lenguaje, no con otros.

Google App Engine cuenta con un plan gratuito de uso que permite un alojamiento máximo de 1 GB y un ancho de banda de 1 GB por día, así como otras características básicas. Si se necesita algo mayor, existen planes con diferentes tarifas de acuerdo a las características necesitadas [3].

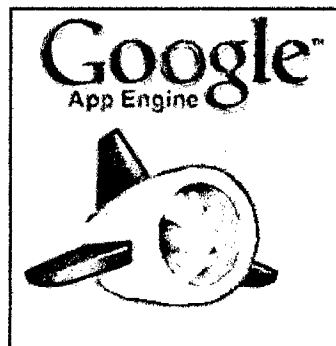


Figura 2.1. Logo de Google App Engine

2.1.6.2. Windows Azure

Windows Azure es la apuesta de Microsoft para aplicaciones en la nube. Es un servicio para desarrollar, alojar y escalar aplicaciones a través de servidores alojados en los data centers de Microsoft.

Ofrece diferentes servicios como SQL Azure para alojamiento de bases de datos, y AppFabric para el alojamiento de aplicaciones .NET y SharePoint entre otras. En Windows Azure actualmente pueden correr aplicaciones construidas con el Framework .NET, y de la misma forma existen SDK oficiales para integrar aplicaciones Java y Python con los servicios alojados en el AppFabric [4].

A diferencia de otros servicios, Microsoft no ofrece planes gratuitos de alojamiento para Windows Azure. Ofrecen un periodo de prueba (trial) con algunas características, orientado a probar y familiarizarse con la plataforma. En cuanto a planes de pago, cuenta con dos modalidades, una con planes de características y precios fijos, y otra flexible en donde se paga por lo que se usa [5].



Figura. 2.2. Logo de Windows Azure2.

2.1.6.3. Amazon Web Services

Amazon Web Services o AWS es una colección de servicios remotos, ofrecidos por Amazon a través de Internet. En estos servicios se pueden alojar sitios o aplicaciones web que se pueden exponer al público o ser usadas por otros desarrolladores.

Entre los servicios más destacados de AWS son el S3 y el EC2. El primero es para alojamiento de aplicaciones, y el segundo ofrece lo que podría llamarse un servidor virtual completo corriendo en la plataforma de Amazon. En el primer caso se cobra por ancho de banda o descargas, mientras que en el segundo el precio varía es basado en la capacidad de procesamiento.

Así mismo, AWS también ofrece alojamiento de bases de datos, sitios de comercio electrónico, backup de información, aplicaciones de Facebook, motores de búsqueda, etc [6].



Figura. 2.3. Logo de AWS3.

2.1.6.4. Engine Yard

Engine Yard es una compañía localizada en San Francisco, California. Se encarga de proveer servicios de desarrollo y alojamiento de aplicaciones en la nube desarrolladas en Ruby onRails [7].

Sus planes de servicios cuentan con varias modalidades. Cuenta con planes de tarifa fija con determinadas características, planes flexibles en donde el cliente puede escoger las características y configuraciones deseadas, así como también planes de pago por capacidad usada [8].

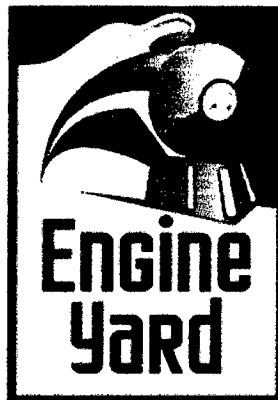


Figura. 2.4. Logo de Engine Yard4.

2.1.6.5 Heroku

Al igual que Engine Yard, Heroku es una compañía focalizada en proveer servicios de plataforma en la nube para Ruby on Rails. Así mismo, es una de las compañías pioneras en ofrecer soluciones basadas en No-SQL, y en la actualidad cuenta con múltiples soluciones.

El precio de sus soluciones se determina basado en la capacidad y recursos que se necesitan. También ofrecen “Addons” para aplicaciones que tienen precios de acuerdo a lo que se requiere. Entre estos “Addons” se cuenta API’s para Facebook, Twitter, Amazon, entre otros [9].

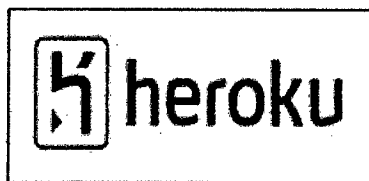


Figura. 2.5. Logo de Heroku5

2.1.7. Arquitecturas para el Desarrollo de Software en la Nube

Cuando se habla de un sistema de computación en la nube, se pueden diferenciar básicamente dos aspectos: en primer lugar, un frontend del lado del usuario, que se compone del computador cliente y de la aplicación requerida para acceder al sistema en la nube. En segundo lugar, se tiene un back end compuesto por varios servidores de aplicaciones y almacenamiento de datos que crean la "nube de servicios".

2.1.7.1. Capas

Software as a Service. Software como servicio (SaaS) se encuentra en la capa más alta y caracteriza aplicaciones que se ofrecen como servicio para ser accedidos a través de una red interna o de internet. Generalmente se requiere sólo un navegador web en la parte del cliente para acceder, lo cual disminuye las necesidades de instalaciones, mantenimiento y soporte. Así mismo, permite el acceso desde cualquier ubicación geográfica en el planeta [10].

Plataform as a Service. La capa del medio, que es la plataforma como servicio (PaaS), es la encapsulación de ambientes de desarrollo en la nube, a los cuales se accede a través de un navegador y en donde se pueden construir y desplegar aplicaciones web en ambientes virtualizados, sin necesidad de tener ningún software instalado en los computadores usados por los desarrolladores, y con las ventajas que brinda el modelo como la transparencia de la configuración de los ambientes usados, la gestión y monitoreo de los ambientes, así como la facturación de acuerdo a la capacidad de uso [11].

Entre los servicios de PaaS más comunes se puede encontrar el AppEngine de Google y la plataforma Windows Azure de Microsoft, así como el muy popular Salesforce y todos sus servicios.

Infrastructure as a Service. La infraestructura como servicio (IaaS) se encuentra en la capa inferior y es la externalización de servicios básicos de infraestructura donde se delega en un proveedor las tareas de administración y soporte de un data center. Con IaaS se tiene servicios de almacenamiento en base de datos, espacio en disco y capacidad de procesamientos flexibles, en entornos virtualizados y muy optimizados de acuerdo a la capacidad de uso [12].

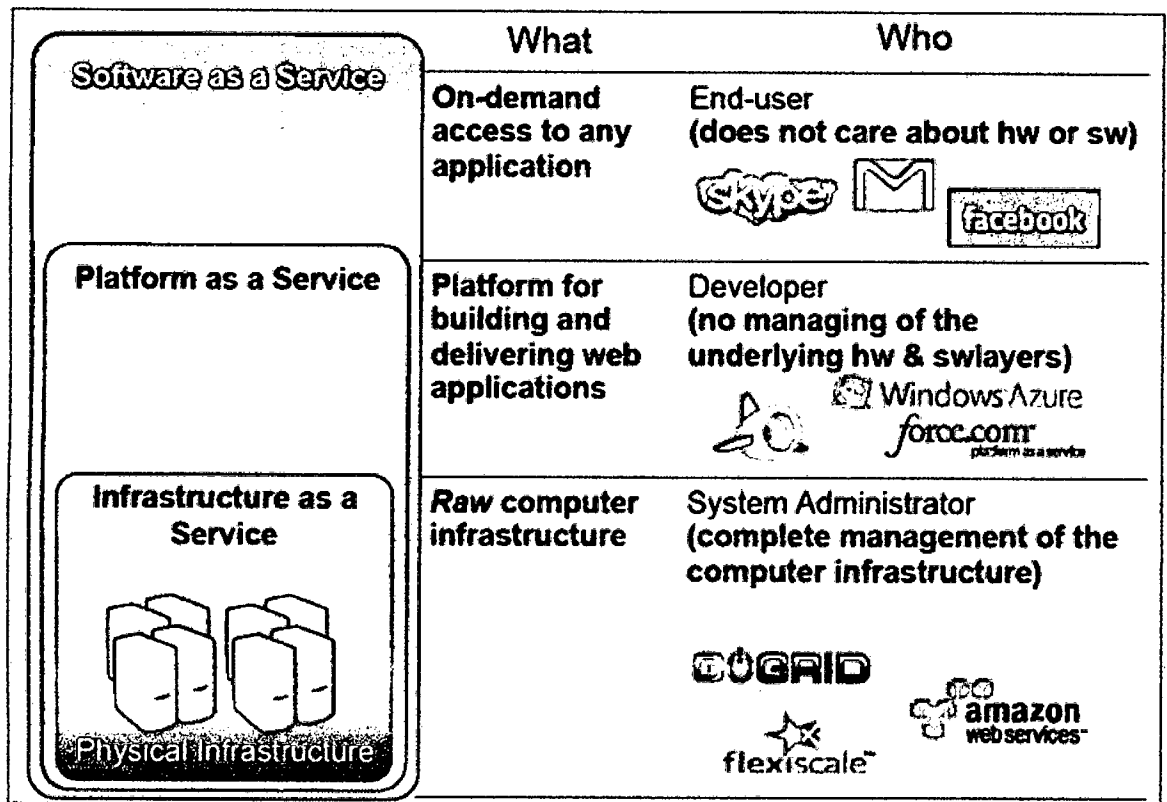


Figura. 2.6. Capas de arquitecturas en la nube6.

2.1.7.2. Arquitecturas

Grid Computing. Grid Computing es una tecnología que permite el procesamiento distribuido sin un servidor centralizado. Esta distribución permite un procesamiento de datos a gran escala, y por esto es una modalidad muy usada para investigaciones o estudios científicos que requieren del procesamiento o cálculo de muchos y complejos datos, los cuales posiblemente requerirían mucho tiempo si se hicieran con un recurso centralizado [13].

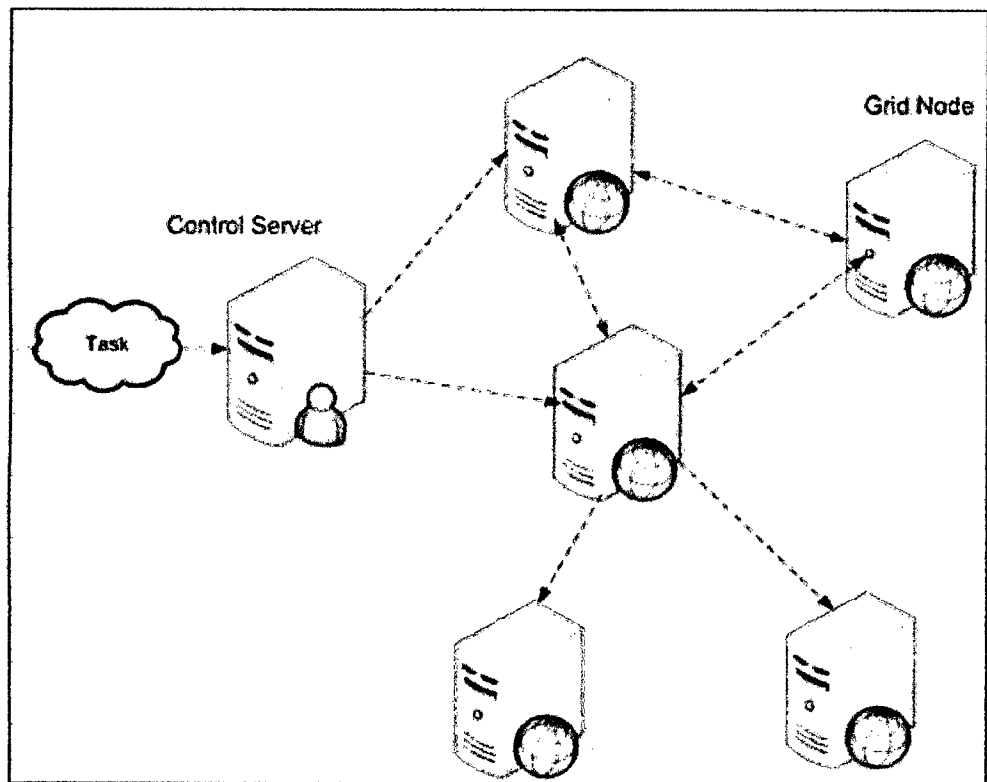


Figura 2.7. Esquema Grid Computing7.

Computación Transaccional. La computación transaccional es la base de la mayoría de aplicaciones comerciales que se encuentran en el mercado, a diferencia del Grid Computing, el cual es más usado a nivel científico.

Una arquitectura de un sistema transaccional se compone de servidores que procesan datos en una transacción y de una base de datos que almacena información y gestiona las relaciones entre los datos que en ella se guardan.

Uno de los temas claves de las aplicaciones transaccionales en la nube es la alta disponibilidad que deben tener, pero por lo general los diferentes proveedores de servicios de Cloud Computing cuentan con estrategias para minimizar dichos fallos [13].

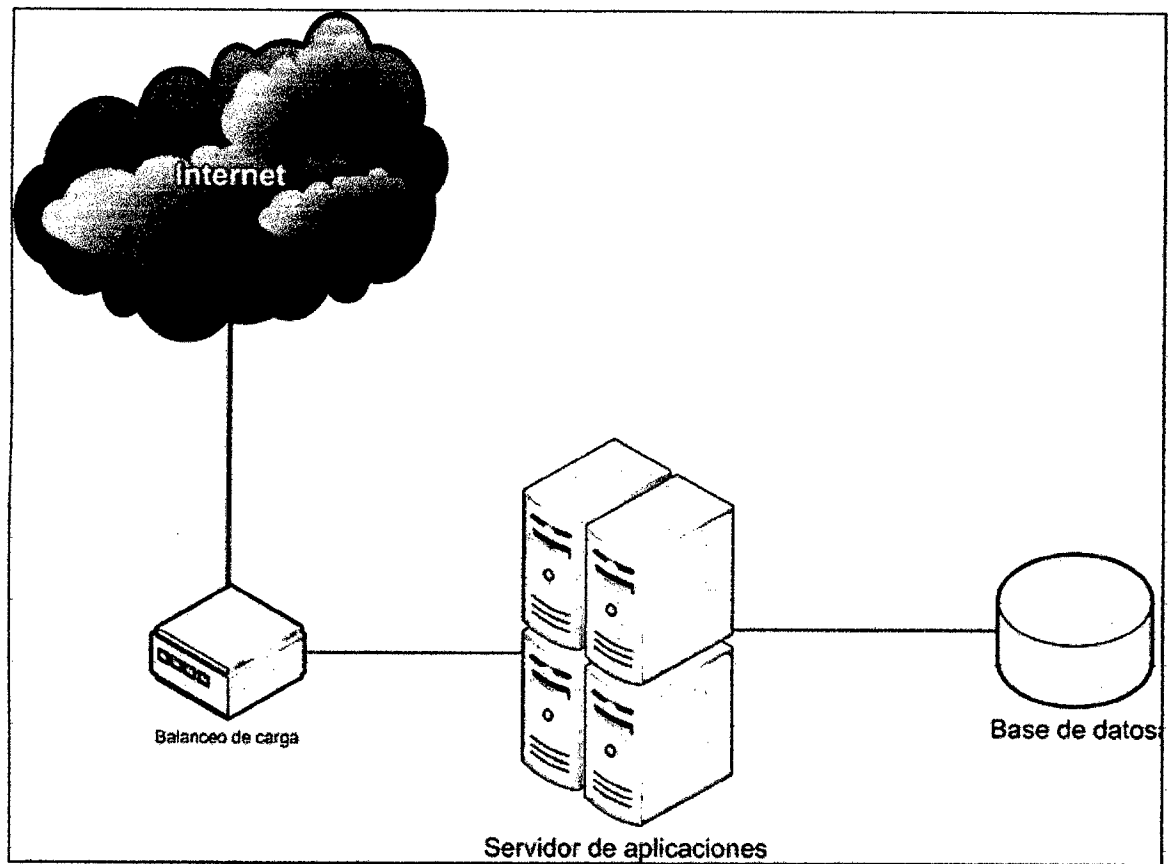


Figura. 2.8. Esquema Computación Transaccional⁸.

2.1.8. Futuro de la computación en la nube

No es difícil imaginar un futuro con una gran proliferación de operadores y usuarios del Cloud Computing. Y teniendo en mente la naturaleza colaborativa de esta modalidad de trabajo, todo indicaría que el futuro de la nube será lo que algunos mencionan como internube, varias nubes interconectadas, aprovechando recursos e información, comunicándose entre sí.

Esto abre puertas a mayores chances de interacción e ilusiona pensar la manera en que se multiplicarán los beneficios para el usuario. Por ejemplo, la posibilidad de cambiar de proveedor libremente a través de migraciones seguras y transparentes, garantizando la continuidad de los procesos. La oferta de servicios crecería todavía más y los costos serían menores.

Para llegar a esos niveles de interoperabilidad hace falta avanzar en varios aspectos, no solamente inversión y coordinación entre los distintos agentes que proveen los servicios. Deberán desarrollarse aún interfaces, reglamentaciones, estándares de seguridad, hardware... y la lista apenas comienza.

2.1.8.1. Tendencias en Torno a la Computación en la Nube.

Construcción de Nubes Privadas: Las Organizaciones y las empresas más grandes están construyendo sus propias nubes privadas. IBM "Blue Cloud" es un ejemplo. Las nubes privadas son enormes centros de datos que poseen información y datos de todo tipo. Quitar las preocupaciones de seguridad acerca de la computación en nube.

Cambio de las Competencias de los Profesionales de las TIC: Con el cambio de paradigma del modelo cliente servidor para el modelo Cloud Computing, los trabajadores de TI necesitarán nuevas habilidades. Ellos ahora no tienen que operar y mantener la infraestructura de TI, pero ahora tendrá que saber cómo extraer y utilizar la mejor información de la nube

Reducción de los Departamentos de TI: Los departamentos de TI de las organizaciones se reducirán enormemente tanto en las infraestructuras y el número de trabajadores. Con el advenimiento de la computación en nube, las empresas ya no necesitan demasiados trabajadores y recursos para hacer todas las tareas relacionadas con TI, porque se van a usar servicios de Cloud Computing.

Disminución Preocupaciones de Seguridad: De hecho, este proceso sólo tiene que tener la madurez y una relación de confianza entre vendedores y clientes. Es posible que las técnicas criptográficas, el nivel de habilidad del personal de los vendedores para garantizar la seguridad y algunos métodos nuevos se puedan idear para resolver este problema.

Servicios de Consultoría Especializada en Cloud Computing: Con el fin de atraer a clientes cada vez más, los vendedores de computación nube también ofrecen servicios profesionales de naturaleza distinta a los servicios actuales. Successforce de Salesforce.com es un buen ejemplo de tales servicios. NetSuite también ofrece estos servicios de consultoría gratuita a sus clientes.

PYMES y Grandes Empresas en la Nube: Algunos proveedores de cloud computing como NetSuite están arrendando el procesamiento y las aplicaciones para las PYMES. Al arrendar software de Microsoft no tiene que adquirir e implementar todo el hardware, software y recursos necesarios para ejecutar sus actividades empresariales.

Personalización de los Recursos de la Nube: En la actualidad la computación en nube se lleva a cabo utilizando las mejores prácticas y estándares. En el futuro con mayor progreso en este paradigma, el Cloud Computing va a funcionar con mayor flexibilidad en casi todos los campos, en lugar de solo CRM y aplicaciones de negocio. Esto hará en los próximos años que se pueda personalizar los servicios de Cloud Computing.

Comercialización de Servicios en la Nube: Las empresas que tienen una enorme infraestructura de TI y recursos razonables construirán sus propias nubes. Por lo tanto, arrendarán a otros proveedores y clientes ganando beneficio extra de eso. Amazon.com ya lo está haciendo desde hace algunos años.

Cloud Computing Traerá Innovación: Hoy en día un gran número de proyectos informáticos innovadores están estancados sólo por la falta de recursos como la infraestructura, los trabajadores calificados, y la energía y costos. Sin embargo, el Cloud Computing va a resolver este problema mediante la introducción de centros de datos gigantes. Ellos serán los principalmente necesarios para el cumplimiento de la necesidad de los clientes de los vendedores, pero no habrá ningún problema para los centros de datos grandes para ayudar en proyectos de computación innovadores y grandes.

Sólo el Navegador Será Necesario: En el futuro sólo necesitará un navegador en el escritorio y a través de él usted podrá utilizar y ejecutar todo lo presente en la nube.

2.1.9. Pruebas de concepto

2.1.9.1. Aplicaciones interesantes en la Nube [14]

Google Docs: Es un programa basado en Web para crear documentos en línea con la posibilidad de colaborar en grupo.

Spotify: El servicio de música streaming, si bien no el único (Deezer, Goear, GrooveShark). Es fácil de usar y con un gran catálogo de canciones donde escoger.

Dropbox: Quizás una de las mejores maneras de tener tus archivos seguros, donde tendremos sincronizados en la nube documentos, vídeos, fotos de nuestro ordenador y donde podremos asignar permisos a las carpetas por si queremos compartir nuestros archivos con otras personas.

Evernote: Se trata de una aplicación que nos permite sincronizar notas en la nube. Posiblemente tengamos el ordenador del trabajo, el de casa, nuestro Smartphone y puede que incluso un iPad. Evernote nos permite tener todas nuestras notas sincronizadas y accesibles, pudiendo compartir esta información con más gente.

Zamzar: Con esta aplicación el proceso de convertir un archivo de un tipo a otro será muy sencillo y sin la necesidad de descargar ningún software.

2.1.9.2 Pruebas de concepto en la Nube

Aplicación: ZOHO

Propósito: Probar las aplicaciones de colaboración que dicha plataforma ofrece como es la centralización de las cuentas de correo, chat integrado, gestión de proyectos, gestión de notas y enlace de interés, etc.

Ruta: <https://www.zoho.com/>

Aplicación: EyesOS

Propósito: Probar el escritorio web (similar al sistema operativo y sus aplicaciones en la Web) que ofrece EyeOS y las funcionalidades propias del escritorio como lo son el manejo del correo electrónico, gestión de archivos y aplicaciones propias de la plataforma.

Ruta: <http://www.eyeos.org/es/>

Aplicación: CRM Dynamics

Propósito: Probar el CRM corporativo de EPM en la Nube, con el fin mostrar la estrategia de operación en la Nube de dicha Organización.

Ruta: <https://epm-spd.crm.dynamics.com/>

2.2. DESCRIPCIÓN GENERAL DE LA TECNOLOGÍA INALÁMBRICA BLUETOOTH

2.2.1 CARACTERÍSTICAS DE BLUETOOTH

La tecnología inalámbrica Bluetooth, ofrece una forma de remplazar cables y enlaces infrarrojos por un enlace de radio universal de corto alcance, con capacidad de crear pequeñas redes PANs.

Bluetooth es un sistema de radio que opera en la banda de frecuencia libre de 2.4GHz. Esta banda de frecuencia está disponible en la mayor parte del mundo. Las bandas, 902 -924 MHz, 2 400 - 2 483,5 MHz y 5 725 - 5 850 MHz son libres para el uso de los sistemas de espectro ensanchado.

Bluetooth soporta un canal de datos asíncrono de hasta tres canales de voz simultáneos. El canal asíncrono soporta comunicación simétrica y asimétrica. En la comunicación asimétrica pueden ser enviados 723.3 Kb/s desde el servidor y 57.6 Kb/s hacia el servidor, mientras que en la comunicación simétrica pueden ser enviados 433 Kb/s en ambas direcciones.

2.2.2 CANALES DE TRANSMISIÓN BLUETOOTH

Bluetooth utiliza 79 frecuencias de radio diferentes con un ancho de banda de 1MHz cada uno, las cuales son accedidas de acuerdo a una secuencia de saltos aleatoria a una velocidad de 1600 saltos/segundo. El canal está dividido en time-slots (ranuras de tiempo), cada ranura corresponde a una frecuencia de salto y tiene una duración de 625 μ s. Cada secuencia de salto en una Piconet está determinada por la dirección del maestro de la misma. En una transmisión, cada paquete debe estar alineado con el inicio de una ranura y puede tener una duración de hasta cinco ranuras de tiempo. Durante la transmisión de un paquete la frecuencia es fija. Para evitar fallas en la transmisión, el maestro inicia enviando en las ranuras

de tiempo pares y los esclavos en las ranuras de tiempo impares. En la Figura 2.18, se puede observar este esquema retransmisión.

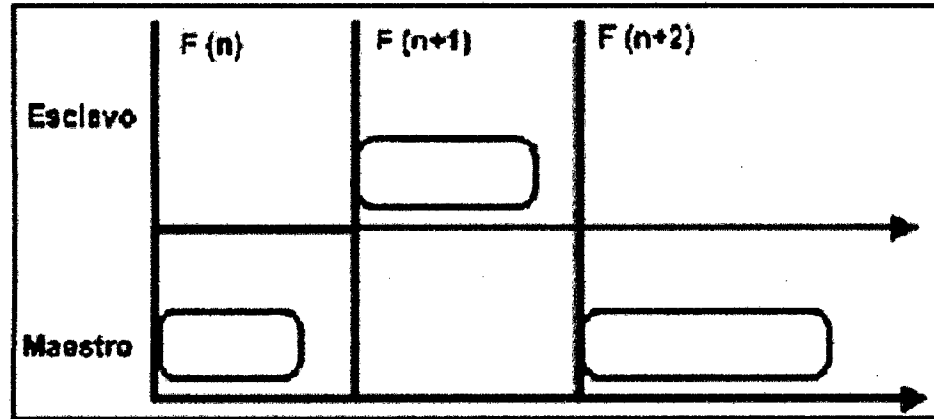


Figura 2.9. Transmisión en una Piconet

2.2.3 DISPOSITIVO BLUETOOTH HC06

El módulo a utilizar para la comunicación Bluetooth entre el sistema Demótico y una Tableta, Celular o PC es el Modulo HC06 que a continuación se describirá.

En la Figura 2.27, 2.28 se muestra un módulo Bluetooth HC-06

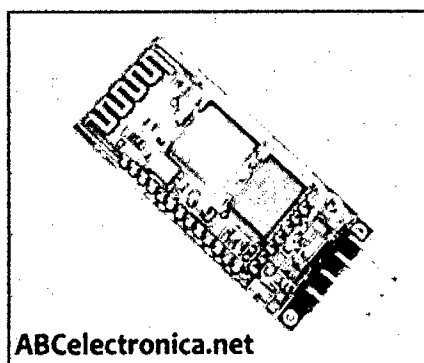


Figura. 2.10 Módulo Bluetooth HC06

2.2.4 Características del Módulo HC06

- Modulo Bluetooth Slave HC-06
- Protocolo bluetooth: Bluetooth especificación V2.0+EDR
- Frecuencia: 2.4Ghz ISM Band
- Rango de baudios ajustable: 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200
- Default: Slave, 9600 baud rate, N, 8,1. Pincode 1234
- Distancia bluetooth: 10 metros
- Tamaño compacto

Una de las ventajas principales del **módulo HC-06**, además de su pequeño tamaño y sus buenas características de transmisión y recepción que le brindan un alcance muy amplio (por tratarse de un sistema local **Bluetooth**), es el **bajo consumo de corriente** que posee tanto en funcionamiento, como en modo de espera, es decir, alimentado con energía, pero sin conexión o enlace a otro dispositivo, por ejemplo, un móvil con SO **Android**.

Otra característica interesante de este módulo es que una vez que ha realizado un enlace con otro dispositivo es capaz de recordarlo en su memoria y no solicita validación alguna ("1234" por defecto), pero si se activa el pin 26 (**KEY**) hacia la tensión de alimentación, esta información se elimina y el **módulo HC-06** solicitará nuevamente la validación del enlace. Otro detalle particular es que su tensión de alimentación de 3,3Volts y su bajo consumo (8mA en transmisión/recepción activa) lo transforman en un dispositivo ideal para trabajar con microcontroladores de la misma tensión de alimentación, logrando de este modo equipos portátiles que pueden ser alimentados durante muchas horas por **baterías recargables o alcalinas AA**, demostrando características excepcionales en aplicaciones médicas, o para actividades recreativas donde la fuente energética debe ser liviana y portátil.

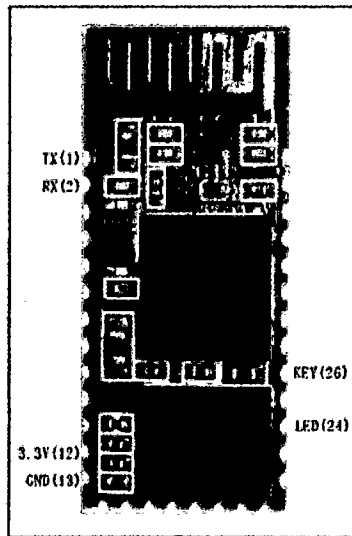


Figura 2.11 Módulo Bluetooth HC06

2.2.5. Conectando el Modulo Bluetooth HC-06 con Microcontrolador

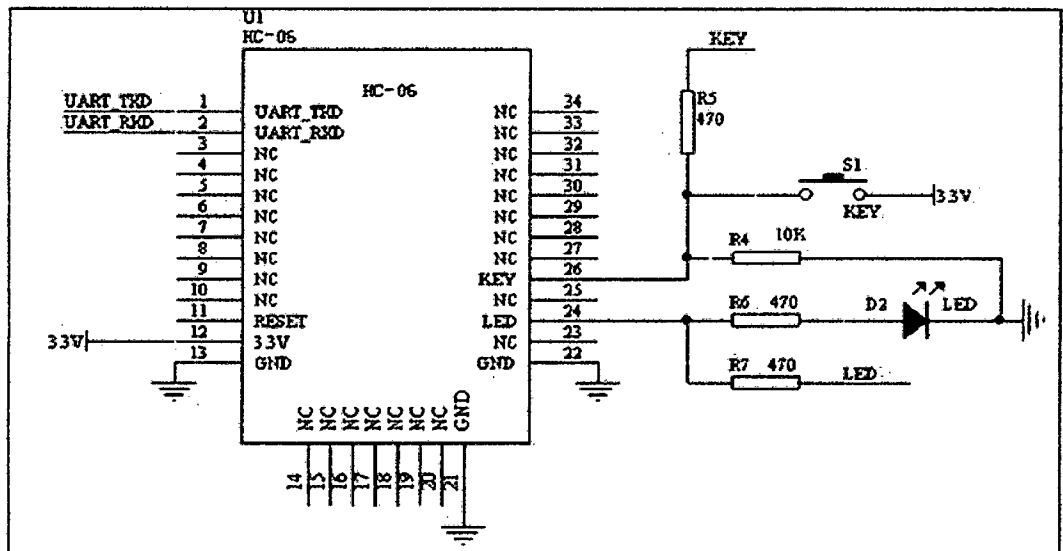


Figura 2.12 Conexiones del Módulo Bluetooth HC06

El primer paso es reconocer que modulo tenemos, para esto debemos conectar la alimentación del módulo a 3.3V, después debemos buscar el dispositivo

bluetooth ya sea con la PC o con un celular, el módulo HC-06 será encontrado con el nombre de “**linvor**”.

Ahora debemos configurar nuestro modulo, el HC-06 se puede configurar por medio de comandos **AT** y los valores que podemos modificar son el nombre del dispositivo, la contraseña (**PIN**) para realizar la conexión y el baudrate. Para que los comandos **AT** funcionen el modulo no debe estar apareado con el dispositivo maestro, debe ser configurado por medio de un microcontrolador o mediante un convertidor USB-serial y la terminal serie en una PC.

Los comandos **AT** disponibles son los siguientes:

AT: Sirve como test de comunicación, responde con **OK**

AT+VERSION: Devuelve la versión del firmware del dispositivo, responde con **OKlinvorV1.5**

AT+NAME: Cambia el nombre del dispositivo, por ejemplo **AT+NAMEdispBT1** responde con **OKsetname** y ahora tendrá el nombre **dispBT1**, el nombre es limitado a 20 caracteres.

AT+PINxxxx: Cambia el pin de seguridad de 4 dígitos, podemos usar **AT+PIN0000** para setear el pin a 0000, responde con **OKsetPIN**, por default viene configurado 1234.

AT+BAUDx: Modifica el baudrate del dispositivo, x puede tomar los siguientes valores

- 1———1200
- 2———2400
- 3———4800
- 4———9600 (Default)
- 5———19200
- 6———38400
- 7———57600
- 8———115200

9———230400
A———460800
B———921600
C———1382400

Debes tomar en cuenta que el baudrate máximo que maneja una PC es de 115200, por lo que si estas configurando tu modulo por medio de esta y escoges un baudrate mayor a 115200 perderás la comunicación completamente con el dispositivo, si esto llega a suceder solo podrás reconfigurarlo por medio de un microcontrolador capaz de manejar tal velocidad mayor a 115200. Si la velocidad no es primordial en tu diseño maneja la velocidad de 9600 por default o en caso necesario la de 115200 como máximo. Para 9600 baudios usamos **AT+BAUD4** y responde **OK9600**

Una vez configurado el dispositivo lo podemos utilizar con un microcontrolador y realizar una comunicación serial de forma transparente.

Nota: Si se está trabajando solo con el chip, debe tomar en cuenta que el fabricante toma el pin TX(1) como la entrada de datos que serán enviados posteriormente por Bluetooth y RX(2) donde salen los datos recibidos por Bluetooth.

2.2.6. Obtener MAC del Módulo Bluetooth HC06

Para realizar una aplicación en una Tablet o Celular en plataforma Android es necesario conocer la MAC del Módulo Bluetooth para lo cual explicaremos los pasos a seguir:

1. En la Computadora Personal se debe tener Instalado Modulo Bluetooth USB y el Software Bluesoleit.

2. Encender o Activar el Modulo Bluetooth

3. En la PC realizar la búsqueda y conexión de clientes para realizar la vinculación luego de un tiempo aparecerá en la pantalla de la computadora el Número de MAC del Módulo HC06 parecido a: **00:11:10:20:02:06** el cual nos

servirá para realizar la aplicación para tabletas o equipos celulares con Sistema Operativo Android. En la figura 2.30 se muestra una pantalla del Software Bluesoleit.

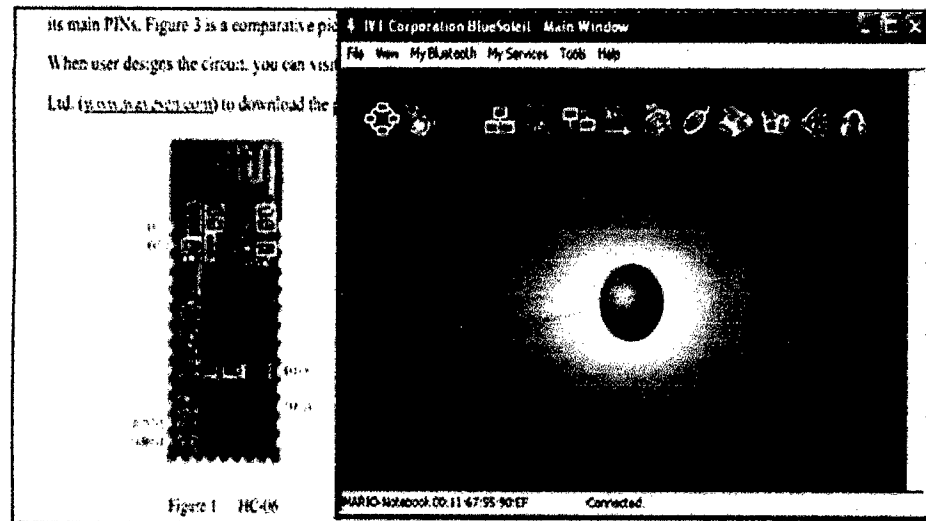


Figura 2.13 Conexión con software Bluesoleit

2.3 DESCRIPCION DE LOS MICROCONTROLADORES

2.3.1. MICROCONTROLADORES PIC

Los microcontroladores PIC (Peripheral Interface Controller) son fabricados por la empresa Arizona Microchip. En la dirección www.microchip.com, dispone de amplia información de todos los productos que fábrica, también podemos descargar gratuitamente los simuladores para dichos microcontroladores.

Los PIC, y en función de los recursos y número de terminales del chip en el que se encuentran integrados, se divide en varias familias.

PIC12CXXX gama baja (8 pines, palabra de programa de 12bits/14bits):

–Bajo consumo.

–Memoria de datos EEPROM.

PIC16C5X, gama baja o clásica (palabra de programa de 12bits):

–Encapsulados de 14, 18, 20y28 pines.

–Óptimo para aplicaciones que trabajan con baterías (bajo consumo).

PIC16CXXX, gama media (palabra de programa de 14bits).

–Convertidores A/D y puerto serie.

–Encapsulados desde 18 a 68 pines.

PIC17CXXX, gama alta (palabra de programa de 16bits).

–Arquitectura abierta, memoria ampliable.

PIC18CXXX, gama alta (palabra de programa de 16bits).

–Conjunto de instrucciones mejorado.

–Detección de bajo voltaje programable (PLVD).

Dentro de cada gama se dispone de una gran variedad de modelos y encapsulados, pudiendo seleccionar el que mejor se adapte a cada proyecto.

Cada uno de los componentes de estas familias comparte el mismo juego de instrucciones, arquitectura interna, interrupciones, etc., aunque puede variar la capacidad de la memoria tanto de programa como de datos, así como el tipo de memoria de programa que utiliza.

Evidentemente según aumentamos en la numeración lo que hacemos es seleccionar dispositivos más complejos que incorporan nuevos recursos, como puede ser conversores AD, comparadores, mayor número de interrupciones, entre otros.

Todos los PIC independientemente de la familia a la que pertenezcan y de los recursos que tenga implementados, poseen unas características que les son comunes

2.3.2 MICROCONTROLADOR PIC16F877A

Arquitectura Interna

El PIC16F877 es un microcontrolador de Microchip de la familia media y al igual que los demás miembros de su familia se caracteriza por:

- Su procesador es tipo RISC (35 instrucciones).
- Velocidad de operación de hasta 20MHz y 200ns por ciclo de instrucción.

- Memoria de programa Flash de 8kx14 palabras.
- Modo sueño de ahorro de energía.
- Vías de entrada de lectura y escritura del procesador.
- Tiene una arquitectura HARVARD.
- Perro guardián Timer (WDT) con propio oscilador RC.
- La protección de código programable.

Recursos especiales

Además de las características anteriormente señaladas, se puede resaltar lo siguiente:

- Timer0: El temporizador/contador de 8bits con pre escalador de 8bits.
- Timer1: El temporizador/contador de 16 bits con pre escalador, puede ser incrementado durante el modo SUEÑO por cristal /clock externa.
- Timer2: El timer/counter de 8bits con registro de período de 8bits, pre escalador y post escalador.
- Dos módulos Capture, Compare, PWM.
- El convertidor multicanal de Analógico a Digital de 10bits.
- Serial síncrono Port(SSP) con SPI e I2C (Master/Slave)
- USART.

Pines y funciones:

El microcontrolador PIC16F877, como se ve en la figura 2.34, dispone de 40 pines. Los *puertos* son el puente entre el microcontrolador y el mundo exterior. Son líneas digitales y otras analógicas que trabajan entre cero y cinco voltios y se pueden configurar como entradas o como salidas.

El PIC16F877 tiene cinco puertos. El puerto A con 6 líneas, el puerto B,C,D con 8 líneas y el puerto E con 3 líneas. Cada pin se puede configurar como entrada o como salida independiente programado por un par de registros diseñados para tal

fin. En ese registro un "0" configura el pin del puerto correspondiente como salida y un "1" lo configura como entrada. Más información se encontrara en el Apéndice.

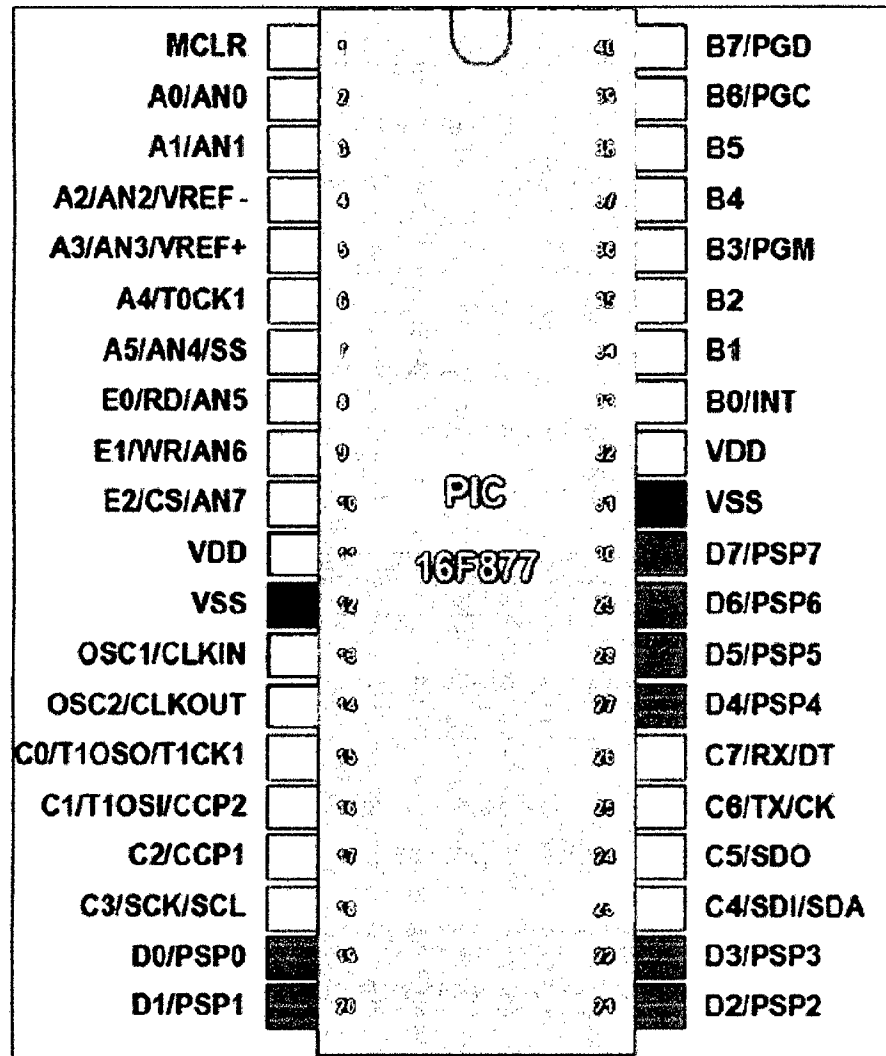


Figura 2.14 Pines del PIC16F877A

2.3.3 MICROCONTROLADOR PIC16F628A

El pic16f628a es un microcontrolador de 8 bit, posee una arquitectura RISC avanzada así como un juego reducido de 35 instrucciones. Este microcontrolador es el remplazo del obsoleto pic16f84a, los pines del pic16f628a son compatibles con el pic16f84a, así se podrían actualizar proyectos que hemos utilizado con el pic16f84a.

En la siguiente figura 2.35 se muestra el diagrama de pines.

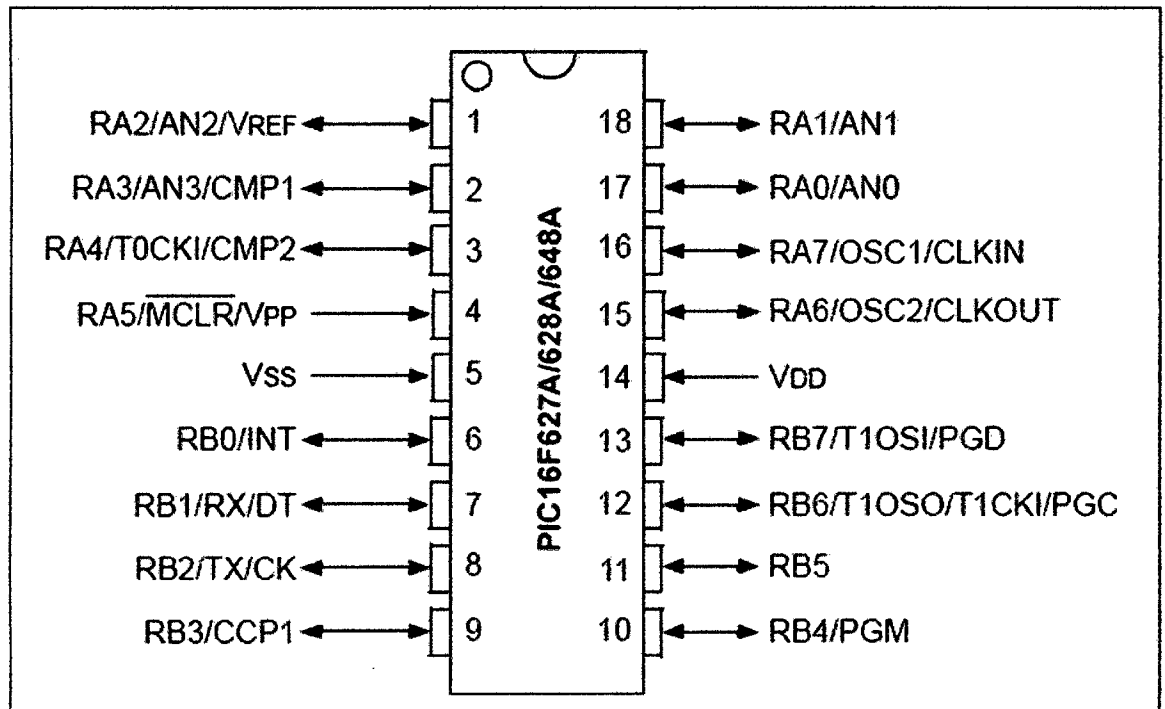


Figura 2.15. Diagrama de Pines del PIC16F628A

Características del PIC16F628A:

CPU De alto rendimiento RISC:

- velocidades de operación de DC - 20 MHz
- Capacidad de interrupción
- pila de 8 niveles
- Modos de direccionamiento directos, indirectos y relativo
- 35 simples instrucciones de palabra:
 - Todas las instrucciones de ciclo único, excepto las de salto

Características Especiales del microcontrolador:

- opciones de oscilador externo e interno:
 - Precisión de fábrica del oscilador interno de 4 MHz calibrada a $\pm 1\%$
 - oscilador de 48 kHz De bajo consumo interno
 - Modo de ahorro de energía en modo sueño
 - resistencias programable pul-ups del PORTB
 - Multiplexado del pin reset / Entrada-pin
 - Temporizador Watchdog con oscilador independiente para un funcionamiento fiable
 - Baja tensión de programación TM In-Circuit Serial (a través de dos pines)
 - Protección de código programable
 - Brown-outreset
 - Power-onReset
 - Power-up Timer y el oscilador de puesta en marcha del temporizador
 - Amplio rango de funcionamiento de tensión (2.0-5.5V)
 - Industrial y amplia gama de temperaturas extendidas
 - Alta durabilidad de la memoria Flash /EEPROM:
 - 100.000 ciclos de escritura Flash
 - 1.000.000 ciclos de escritura EEPROM
 - 40 años de retención de datos
- De baja potencia Características:
- Corriente en espera:

- 100 nA@2.0V, típico
- Corriente de funcionamiento:
 - 12µA @ 32 kHz, 2,0 V, típica
 - 120µA @ 1 MHz, 2,0 V, típica
- Temporizador Watchdog actual:
 - 1µA@2.0V, típico
- Timer1 oscilador actual:
 - 1.2µA @ 32 kHz, 2,0 V, típica
- Doble velocidad del oscilador interno:
 - Tiempo de ejecución seleccionable entre 4 MHz y de 48 kHz
 - 4µs despertar de un sueño, 3.0V, típico

2.4. COMPILADORES

Un compilador es un programa que permite traducir el código fuente de un programa en lenguaje de alto nivel, a otro lenguaje de nivel inferior (típicamente lenguaje de máquina). De esta manera un programador puede diseñar un programa en un lenguaje mucho más cercano a cómo piensa un ser humano, para luego *compilarlo* a un programa más manejable por una computadora.

2.4.1. COMPILADOR PICBASIC PROCOMPILER

PIC BASIC PRO es uno de los compiladores más sencillos de usar, y por esa razón uno de los más populares. Gracias a la versión 8 de MPLAB de Microchip, es posible subir este compilador a uno de los entornos más completos para simulación de código para microcontroladores.

PicBasic Pro de Micro Engineering Labs Inc. (<http://www.melabs.com/>) es uno de los más conocidos. Este poderoso compilador pone al alcance del usuario potentes instrucciones para comunicación serie, matemática de 16 bits, mediciones de Sensores analógicos, PWM, sonido, y muchísimas más. Además de general los

files “hex” (Figura 2.36) y también es capaz de generar los files “asm”. De tal manera que sí se pueden hacer modificaciones de bajo nivel.

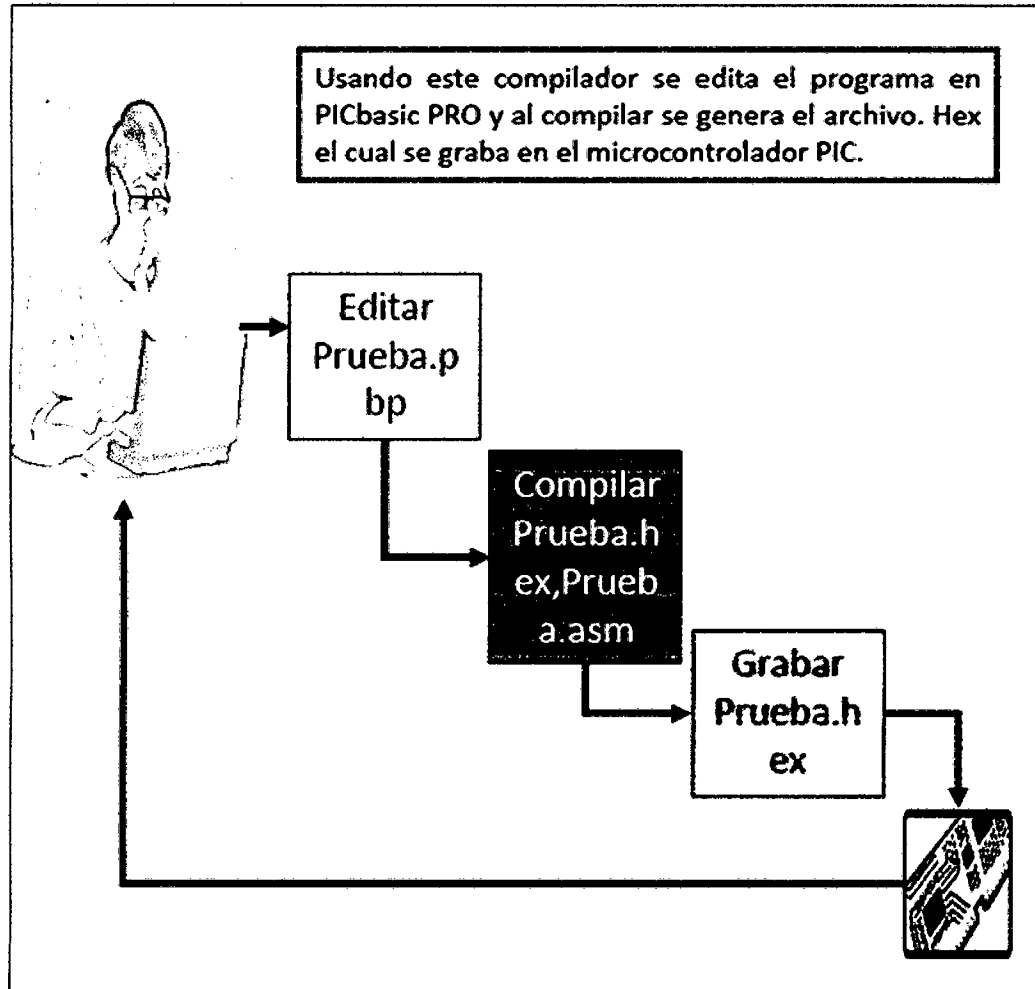


Figura 2.16 Generación de Archivo “Hex” con PBP

2.5 SISTEMA OPERATIVO ANDROID

El sistema operativo Android fue creado originalmente para ser usado en dispositivos móviles de tercera generación pero luego se comenzó a utilizar en otros dispositivos móviles como notebooks, i-pods, mp3s, tablets y ahora se pueden ver hasta en electrodomésticos caseros como lavadoras y microondas.

En el año 2010, Los teléfonos inteligentes con Android ocuparon el primer lugar en ventas en los Estados Unidos. En la actualidad, Android ostenta alrededor del

40% de cuota de mercado a escala mundial en lo que se refiere a teléfonos móviles de tercera generación situándose por delante de Symbian OS e iOS.

Una de las cosas que hacen de Android un sistema operativo para teléfonos móviles distinto a otros como el iOS y Windows Phone es que se desarrolla de forma abierta y se puede ingresar al código fuente así como al listado de incidencias, desde donde podemos ver problemas no resueltos y reportar problemas nuevos. En este artículo conoceremos más acerca de este sistema operativo.



Figura 2.17 Logo Oficial de ANDROID

En el año 2005, Google compro la compañía Android Inc. Lo único que se sabía en ese momento era que Android desarrollaba un software para teléfonos móviles dando a entender que Google tenía en mente ingresar al mercado de la telefonía celular.

El 5 de diciembre del 2007, durante la inauguración de la “Open Handset Alliance” se estrenó Android como una plataforma de soporte para equipos móviles creados en la versión 2.6 del kernel de Linux. Desde entonces el avance de Android ha sido ascendente y hoy es uno de los sistemas operativos para móviles más usados del mundo.

CARACTERÍSTICAS:

Entre las principales características que posee Android podemos mencionar:

- **Conectividad:** Soporta tecnologías de conectividad como Wi-Fi, Bluetooth, GSM/EDGE, UMTS, Wi-MAX y otras más.
- **Mensajería:** Las formas más comunes como SMS y MMS están disponibles además del servicio PushMessaging de Android.
- **Video llamada:** Por medio de la versión HoneyComb, Android soporta video llamadas a través de Google Talk.
- **Soporte multimedia:** Puede soportar los formatos más conocidos como JPEG, MP3, MPEG-4, WAV, además de otros como WebM, H.263 y H.264.
- **Multi- táctil:** Android cuenta con soporte base para equipos móviles con pantallas multi –táctiles.
- **Almacenamiento:** Posee una base SQLite, la cual es utilizada para almacenamiento de datos.

PROGRAMACIÓN:

El desarrollo de aplicaciones para Android es sumamente sencillo y lo único que se necesita es un conocimiento básico de Java y poseer el kit de desarrollo de software provisto por Google. Este kit puede ser descargado completamente gratis.

DISPOSITIVOS:

Android es el sistema operativo con mayor presencia en dispositivos móviles como notebooks, tablets, i-pods, reproductores de mp3 y más. Android es uno de los sistemas operativos que ha logrado establecerse firmemente en el mercado en poco tiempo y además es el sistema con el mayor potencial de desarrollo en el mundo de la telefonía móvil.

Se calcula que en el actualidad hay más 400 000 aplicaciones para Android y que diariamente se activan alrededor de 500 000 equipos móviles.

2.6 PROGRAMACION EN APPINVENTOR

Google App Inventor fue una aplicación de Google Labs para crear aplicaciones de software para el sistema operativo Android. De forma visual y a partir de un conjunto de herramientas básicas, el usuario enlazando una serie de bloques para crear la aplicación. El sistema es gratuito y se puede descargar fácilmente de la web. Las aplicaciones fruto de App Inventor están limitadas por su simplicidad, aunque permiten cubrir un gran número de necesidades básicas en un dispositivo móvil.

Con Google App Inventor, se espera un incremento importante en el número de aplicaciones para Android debido a dos grandes factores: la simplicidad de uso, que facilitará la aparición de un gran número de nuevas aplicaciones; y el Android Market, el centro de distribución de aplicaciones para Android donde cualquier usuario puede distribuir sus creaciones libremente.

2.6.1 HISTORIA

La aplicación se puso a disposición el 12 de julio de 2010 y está dirigida a personas que no están familiarizadas con la programación con ordenadores. En la creación de App Inventor, Google se basó en investigaciones previas significativas en informática educativa. Fue creada a mediados del 2009 el profesor Harold Abelson del MIT.

Antes de salir al mercado se ha probado en diferentes centros educativos y la han utilizado desde niños de 12 años hasta licenciados universitarios sin nociones de programación.

A principio de agosto de 2011 **Google anunció que ya no mantendría esta aplicación**, pero que la haría código libre destinado a la educación. Una semana después el **Instituto Tecnológico de Massachusetts (MIT)**, una institución de educación superior privada situada en Cambridge, Massachusetts (EE.UU.), **anunció que se haría cargo del proyecto**. Ver web.

El 31 de diciembre de 2011 App Inventor de Google dejó de funcionar. El 4 de marzo de 2012, el Instituto Tecnológico de Massachusetts (MIT) volvió a poner el proyecto en Internet.

2.6.2 CARACTERISTICAS

El editor de bloques de la aplicación utiliza la librería Open Blocks de Java para crear un lenguaje visual a partir de bloques. Estas librerías están distribuidas por Massachusetts Institute of Technology (MIT) bajo su licencia libre (MIT License). El compilador que traduce el lenguaje visual de los bloques para la aplicación en Android utiliza Kawa como lenguaje de programación, distribuido como parte del sistema operativo GNU de la Free Software Foundation.

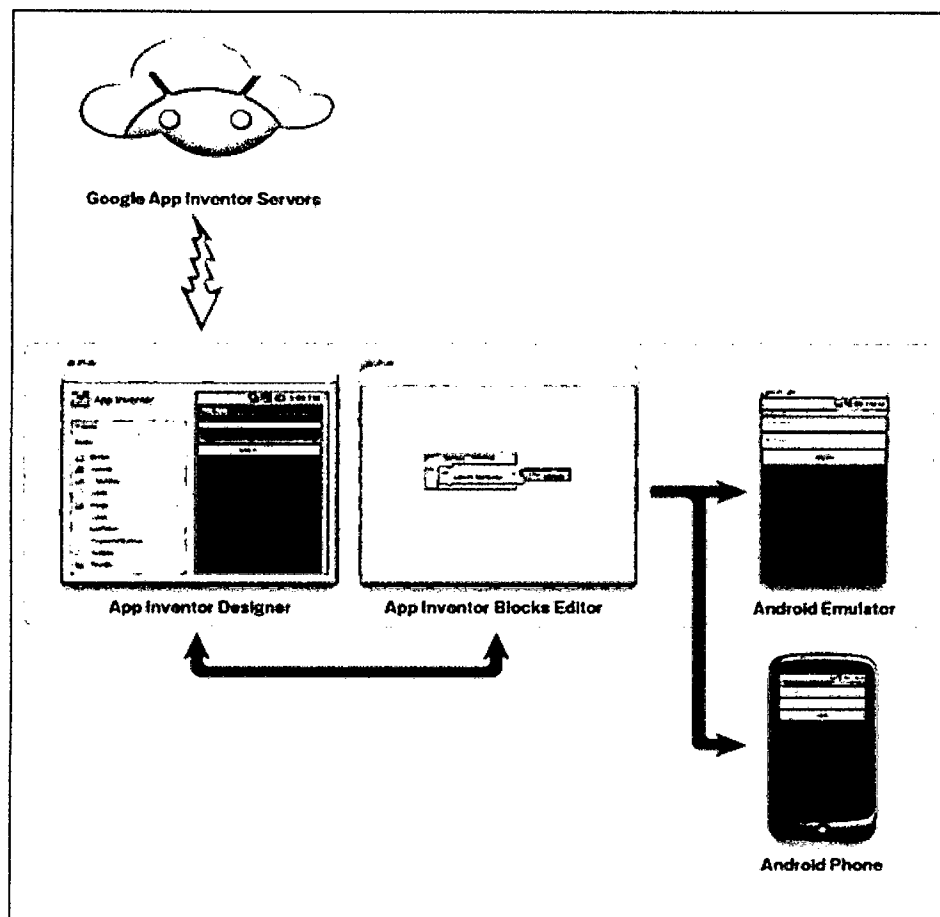


Figura 2.18. Programando en APPINVENOR

El App inventor **no es un programa que se baja al ordenador**. Funciona en **cloud computing**, el programa funciona mediante la conexión a Internet, se trabaja con el programa conectado directamente al servidor de App inventor. Es como una página web y en ella realizas las operaciones.

Para que funcione se necesita **tener instalado JAVA**. También asociar el correo del usuario a Google, esto se realiza inmediatamente sin dificultad.

Para crear una aplicación con App Inventor hay que realizar los siguientes pasos:

- El diseño de la aplicación, en la que se seleccionan los componentes para su aplicación.
- El editor de bloques, donde se ira escogiendo los bloques que sean necesarios según la aplicación
- La aplicación aparecerá paso a paso en la pantalla del celular o tableta a medida que se añada piezas a la misma.
- Cuando se termina, se empaqueta la aplicación y se produce una aplicación independiente para instalar.
- Si no se tienes un celular o tableta Android, se puede construir las aplicaciones utilizando el emulador de Android, el software que se ejecuta en la computadora y se comporta como el celular.

El entorno de desarrollo de App aplicación es compatible con Mac OS X, GNU / Linux y sistemas operativos de Windows, y varios modelos de teléfonos Android populares. Las aplicaciones creadas con App Inventor se pueden instalar en cualquier teléfono Android.

En el diseñador escogeremos los componentes que vayamos a utilizar en nuestra aplicación, según nos interese.

2.7. Sensores de gas

Un sensor es un dispositivo para detectar y señalar una condición de cambio, tales como magnitudes físicas o químicas y transformarlas en variables eléctricas. Los sensores posibilitan la comunicación entre el mundo físico y los sistemas de medición y/o control, tanto eléctricos como electrónicos, utilizándose extensivamente en todo tipo de procesos para propósitos de monitoreo, medición, control y procesamiento. Para el diseño de este proyecto se utilizará un sensor de gas.

Los sensores de gases generalmente constan de elementos físicos sobre los cuales se deposita una capa activa capaz de absorber y/o interactuar con ciertos gases. Estas capas actúan como elementos detectores de gases y dan una gran información de los elementos químicos que conforman el medio ambiente.

El sensor de gas a utilizar es:

2.7.1. Módulo Sensor de humo y gases inflamables MQ-2:

Este sensor detecta las concentraciones de humo y gases inflamables (metano, butano, hidrógeno entre otros) en el aire. El sensor puede medir concentraciones o fugas de gas inflamable de 300 a 10.000 ppm, puede operar a temperaturas de -20 a 50 C° y consume menos de 150mA a 5V.

Este sensor es de alta sensibilidad y con un tiempo de respuesta rápido, consta de 4 pines macho para su conexión (pin de alimentación, pin de ground, pin de salida analógica y pin de salida digital).

Dimensiones: 32mmm X 22mm X 24mm

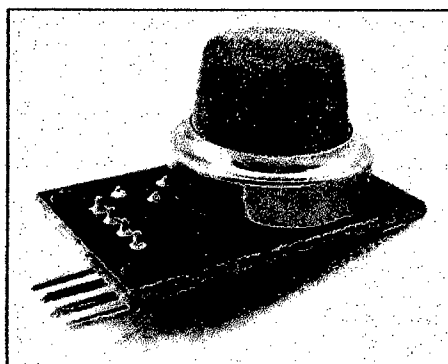


Figura 2.19 Sensor MQ2

La unidad de medida de los sensores MQ-2 es ppm (Partes por millón).

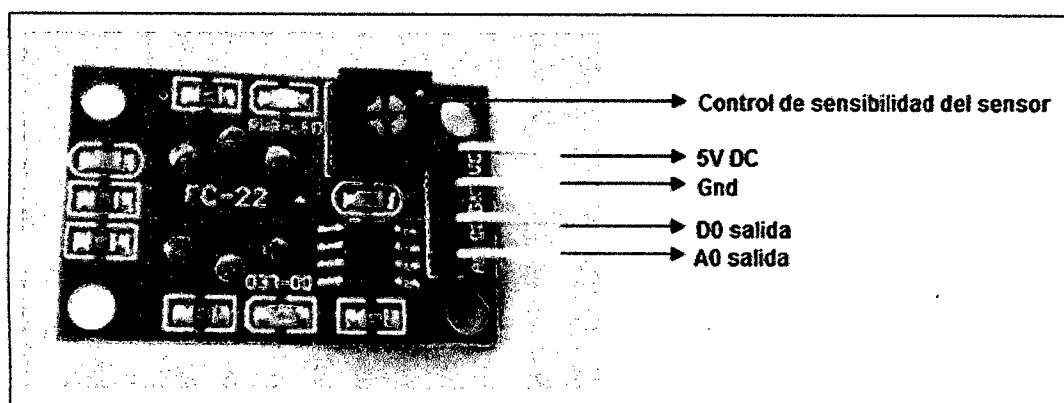


Figura 2.20 Sensor MQ2

Aplicaciones:

- Detección de escape del gas para los restaurantes/cocinas.
- Control de calidad del aire para los edificios/las oficinas.
- Sistemas de vigilancia del CO para los aparcamientos.
- Sistemas de detección de gas para los edificios comerciales, plantas industriales.
- Sistemas de detección de fuego.

2.8. Componente TinyWebDB de AppInventor

TinyWebDB es un componente App Inventor que permite almacenar datos de forma persistente en una base de datos en la web. Debido a que los datos se almacenan en la web en lugar de un teléfono en particular, **TinyWebDB** se puede utilizar para facilitar la comunicación entre los teléfonos y las aplicaciones (por ejemplo, juegos para varios jugadores).

De forma predeterminada, **TinyWebDB** usa un servicio de prueba proporcionado por App Inventor, esta es la dirección: <http://appinvtinywebdb.appspot.com/>.

Este servicio es útil para las pruebas, y es compartida por todos los usuarios de App Inventor, y tiene un límite de 1.000 entradas. Luego los datos serán sobrescritos con el tiempo.

Para la mayoría de las aplicaciones con web necesitaremos crear páginas web personalizadas y que no se comparta con otras aplicaciones, no se necesita ser un programador para hacerlo - sólo tenemos que seguir las instrucciones que se indican a continuación:

- Descargar App Engine para Python en <http://code.google.com/appengine/>. Después de instalarlo, ejecute el GoogleApp Engine Launcher haciendo clic en su icono.
- Descargar el código de ejemplo de <http://beta.appinventor.mit.edu/learn/reference/other/tinywebdbassets/customtinywebdb.zip>. Se trata de un archivo zip que contiene el código fuente de su servicio personalizado Tinywebdb.
- Descomprimir el archivo zip descargado. Se creará una carpeta con el nombre **customtinywebdb**. Puede cambiarle el nombre si lo desea.
- En el GoogleAppEngineLauncher, seleccione **File | Add Existing Application**. Examinar para establecer la ruta a la carpeta que acaba de **customtinywebdb** descomprimido. A continuación, haga clic en el botón Ejecutar. Esto pondrá en marcha un servicio web de prueba que se ejecuta en su máquina local.

- Puede probar el servicio con la apertura de un navegador y entrar en "local host: 8080" como la dirección URL. Usted verá la interfaz de la página web para su servicio web. El objetivo final de este servicio es para comunicarse con una aplicación móvil creada con App Inventor.

Su aplicación no está aún en la web, y por lo tanto aún no accesible para una aplicación de App Inventor. Para llegar allí, lo que necesita para subirlo a los servidores de Google App Engine.

- En el GoogleAppEngineLauncher, seleccione **Dashboard**. Ingrese su información de la cuenta de Google y usted será llevado a un panel de App Engine.
- Seleccione **Create An Application**. Tendrá que especificar un identificador de aplicación única a nivel mundial. Recuerde que el identificador de la aplicación ya que lo necesitará más adelante. Proporcione un nombre para su aplicación y haga clic en **Create Application** para enviar. Si el identificador es único, ahora tiene una nueva aplicación, vacía en los servidores de Google.
- Abra un editor de texto en su computadora local y abra el archivo **app.yaml** dentro de la carpeta **customtinywebdb** ha descomprimido. Modificar la primera línea para que la aplicación coincide con el identificador de la aplicación se establece en Google.
- En GoogleAppEngineLauncher, elija **Implementación** y siga los pasos para la implementación de su aplicación.

Para ver si la aplicación se está ejecutando en la web. En un navegador, introduzca myapp.appspot.com. La aplicación debería ser el mismo que cuando se ejecutó en el servidor de prueba local. Sólo ahora, es en la web y se puede acceder desde el App Inventor para Android.

Las aplicaciones de App Inventor pueden almacenar y recuperar datos a través de su nuevo servicio. Simplemente haga lo siguiente:

Arrastre en un componente TinyWebDB en el Diseñador de componentes.

Modifique la propiedad Service Url del valor predeterminado <http://appintinywebdb.appspot.com/> a su servicio web.

Las operaciones StoreValue (bloques) almacenará los datos a su servicio, y las operaciones GetValue recupera de su servicio.

2.9 La Industria Alimentaria

La **industria alimentaria** es la parte de la industria que se encarga de todos los procesos relacionados con la cadena alimentaria. Se incluyen dentro del concepto las fases de transporte, recepción, almacenamiento, procesamiento, conservación, y servicio de alimentos de consumo humano y animal. Las materias primas de esta industria consisten principalmente de productos de origen vegetal (agricultura), animal (ganadería) y fúngico (perteneciente o relativo a los hongos). Gracias a la ciencia y la tecnología de alimentos el progreso de esta industria se ha visto incrementado y nos ha afectado actualmente en la alimentación cotidiana, aumentando el número de posibles alimentos disponibles en la dieta. El aumento de producción ha ido unido con un esfuerzo progresivo en la vigilancia de la higiene y de las leyes alimentarias de los países intentando regular y unificar los procesos y los productos.

2.9.1. Los Procesos de la Cadena Alimentaria

Los procesos de producción pueden clasificarse en seis diferentes: transporte de alimentos, recepción de alimentos, almacenamiento de alimentos, procesamiento de alimentos, la conservación de alimentos y el servicio de alimentos.

Transporte de alimentos

En este proceso se incluyen todos los procedimientos que buscan proteger la inocuidad de los alimentos durante su traslado desde el lugar de producción al de almacenamiento o procesamiento. En este campo se aplican conceptos de ingeniería para mantener las temperaturas adecuadas y evitar la contaminación ambiental.

Recepción de Alimentos

Es sólo una parte del eslabón en la cadena alimentaria la recepción de mercancías es punto crítico de control en los sistemas de aseguramiento de la calidad visto que si se reciben materias primas descompuestas muy difícilmente se puede recuperar la salubridad. Se aplican en esta fase criterios de aceptación y rechazo de mercancías muy frecuentemente decididas por el juicio profesional de los inspectores de calidad.

Almacenamiento de Alimentos

El almacenamiento de alimentos está orientado a mantener en el tiempo las características propias de los alimentos a fin de que sean mantenidos inventarios para la producción. Se estudia la estacionalidad de ciertos productos alimenticios para controlar la rotación. Se controlan las temperaturas de almacenamiento y la humedad relativa en los lugares de almacenamiento para evitar deterioro acelerado. Generalmente suelen emplearse para el almacenamiento en silos, almacenes acondicionados al tipo de industria específico (herméticos, al aire libre, refrigerados, etc), cámaras frigoríficas, etc y etc

Procesamiento de alimentos

El procesamiento de alimentos es tan variado como la cantidad de alimentos que existen. Cada alimento tiene su procesamiento y la cantidad o complejidad de los procedimientos varía según el tipo de alimento. Normalmente el procesamiento de alimentos se diagrama en flujogramas de proceso y su control depende de los sistemas de aseguramiento de la calidad

Conservación de Alimentos

Los procesos habituales de la conservación de alimentos, tienen como objeto la transformación inicial del alimento para la obtención de otro producto distinto y transformado que presente un mayor tiempo de vida útil. Algunos de los procesos de conservación son: - Salado. - Deshidratación. - Refrigeración - Congelación. - Pasteurización. - Esterilización – Acidificación

Servicio de Alimentos

Dentro de la industria alimentaria se incluyen los establecimientos que se encargan de preparar alimentos para su servicio y consumo. En estos establecimientos se aplican técnicas gastronómicas que se deben controlar según las normas de la higiene de alimentos. Dentro de este grupo se incluyen a los restaurantes, comedores públicos, comedores escolares, Comedores industriales, hoteles, clínicas, hospitales, cruceros y ventas ambulantes de alimentos.

2.9.2. Sectores de la industria

Generalmente la industria alimentaria se ha considerado como un conjunto de establecimientos que se agrupan según el tipo de alimentos, entre estos tenemos los siguientes a saber:

- Industria cárnica
- Industria pesquera y de transformación de pescado
- Sector agropecuario
- Tecnología de frutas y hortalizas
- Aceite
- Industria láctea
- Productos Alimentación Animal
- industria de la Panificación
- Azúcar
- Cacao
- Vinos
- Cerveza
- Bebidas alcohólicas
- Agua
- Otros productos diversos

2.9.3. VARIABLES UTILIZADAS EN LA INDUSTRIA ALIMENTARIA

En la industria alimentaria, las variables que se necesita medir con mayor frecuencia son el gasto o cantidad de materia procesada por cantidad de tiempo, la concentración, la presión, la temperatura, el trabajo efectuado, la potencia y variables afines. El sistema de unidades más utilizado es el Sistema Internacional de Unidades, cuyas principales unidades son:

- Longitud > metro (m)
- Masa > kilogramo (kg)
- Tiempo > segundo (s)
- Temperatura > Kelvin (°C)
- Fuerza > Newton ($N = \text{Kgm/s}^2$)
- Energía > Julio ($J = \text{Nm} = \text{kgm}^2/\text{s}^2$)
- Potencia > Vatio ($W = \text{J/s} \text{ ó } \text{kgm}^2/\text{s}^3$)
- Presión > Pascal ($\text{Pa} = \text{N/m}^2$)

Composición:

Las sustancias procesadas en la industria alimentaria casi nunca son puras y los productos resultantes por lo general son mezclas; la composición se mide según fracción másica y molar (cantidad de masa de una sustancia expresada en kg o kg mol por unidad de masa total), relación masa y molar (cantidad de masa de una sustancia expresada en kg o kg mol por unidad de masa libre de la sustancia por medir), molaridad (número de gramos mol de una sustancia contenidos en un litro de solución), modalidad (número de gramos mol de una sustancia contenidos en un litro de solvente) y normalidad (número de gramos equivalentes de una sustancia contenidos en un litro de solución).

Densidad: Cantidad de masa por unidad de volumen ($d = m/V$).

Densidad relativa: Relación de la densidad de una sustancia a la densidad del agua ($d_r = d_{\text{sustancia}}/d_{\text{H}_2\text{O}}$).

Fuerza: Aquello capaz de modificar la velocidad de un cuerpo (1a ley de Newton) ($F = ma$)

Peso: Fuerza con la cual la tierra atrae a los cuerpos hacia su centro. **Peso**

específico: Peso de la unidad de volumen de una sustancia ($P_e = F/V$)

Presión: Toda fuerza ejercida perpendicularmente sobre un área ($P = F/A$).

Presión hidrostática: La presión en el seno de un líquido es igual a la altura del mismo sobre ese punto multiplicada por el peso específico del líquido {Presión hidrostática= $P_e(h)$ }

Temperatura: Medida del nivel energético de las sustancias, para medirla se usan escalas, siendo las más comunes la centígrada y la Fahrenheit { $^{\circ}F = ^{\circ}C (1.8) + 32$ }

Energía: Todo aquello capaz de producir un trabajo. **Energía potencial:** debida a la posición que guarda un cuerpo con respecto a otro ($EP = mgh$; donde m =masa, g =aceleración de la gravedad y h =altura).

Energía cinética: debida a la velocidad que tiene un cuerpo { $EC = m (v)^2/2$ }

Trabajo: Producto de la fuerza por una distancia ($t = Fd$).

CAPITULO III

DISEÑO HARDWARE Y SOFTWARE DEL SISTEMA

3.1. Diagrama de Bloques del Sistema.

El diagrama de bloques del Sistema el que se muestra en la figura 3.1, en el cual se puede apreciar que se pretende diseñar un sistema, que permita leer o monitorear las diferentes variables de una planta y su proceso. Estos datos serán almacenados en los servidores de Google, para lo cual se utilizara un Tableta o celular con sistema Operativo Android, el cual no servirá como medio de conexión hacia el Internet y estos servidores.

Luego estos datos podrán ser bajados o leídos remotamente usando computadoras o tabletas conectadas a Internet.

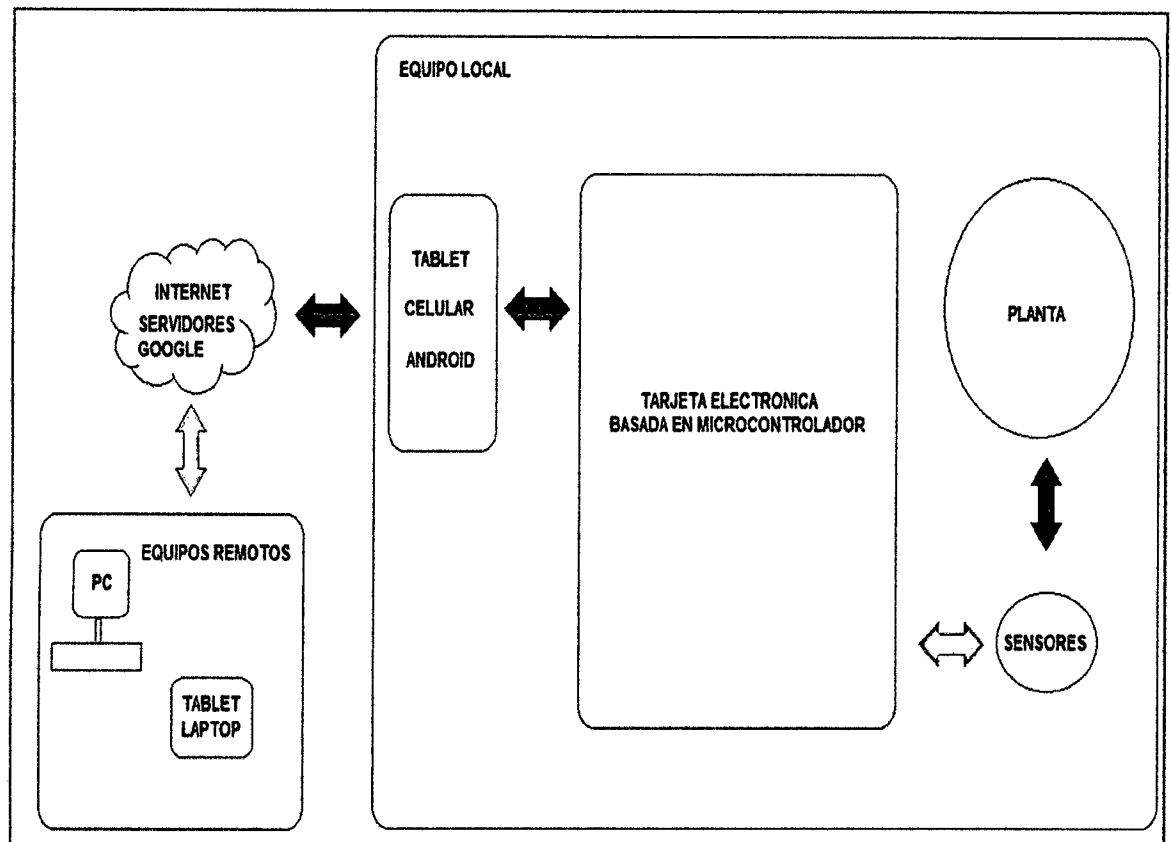


Figura 3.1 Diagrama de Bloques del Sistema

En la Figura3.2 se muestra un diagrama de bloques más completo del sistema que se plantea diseñar

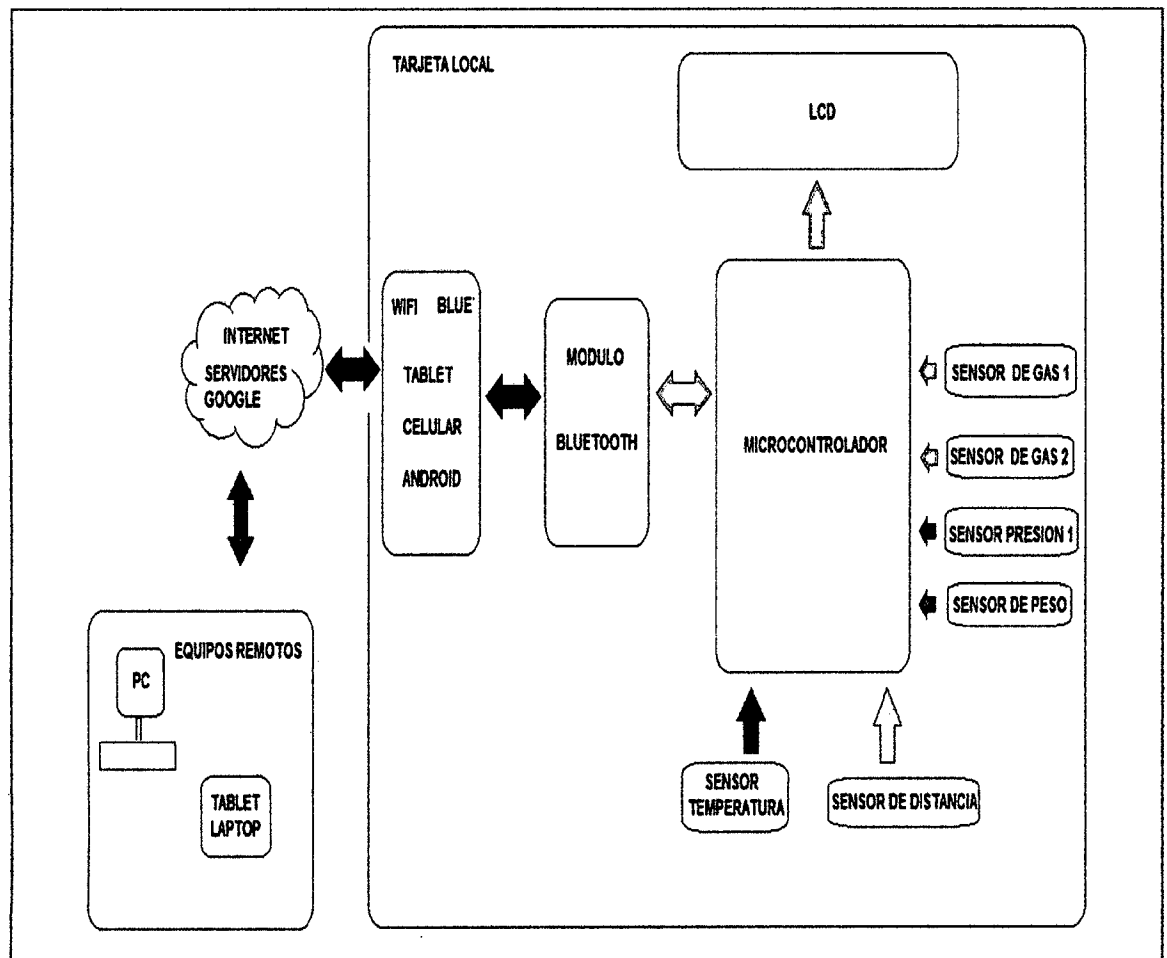


Figura 3.2 Diagrama de Bloques del Sistema

3.2. Diagrama Electrónico de la Tarjeta para sistema

En la Figura 3.3.y 3.4 se muestra el diagrama de la tarjeta electrónica para sistema que se plantea.

Esta tarjeta está basada en el microcontrolador PIC16F877A., con sensores de: Temperatura como LM35DZ, Sensor de Distancia como el HCSR04, sensores de

GAS MQ7, MQ2, y potenciómetros para simulación de otros sensores como sensor de presión y Fuerza. Para la comunicación con tableta o celular se utiliza el modulo Bluetooth HC06.

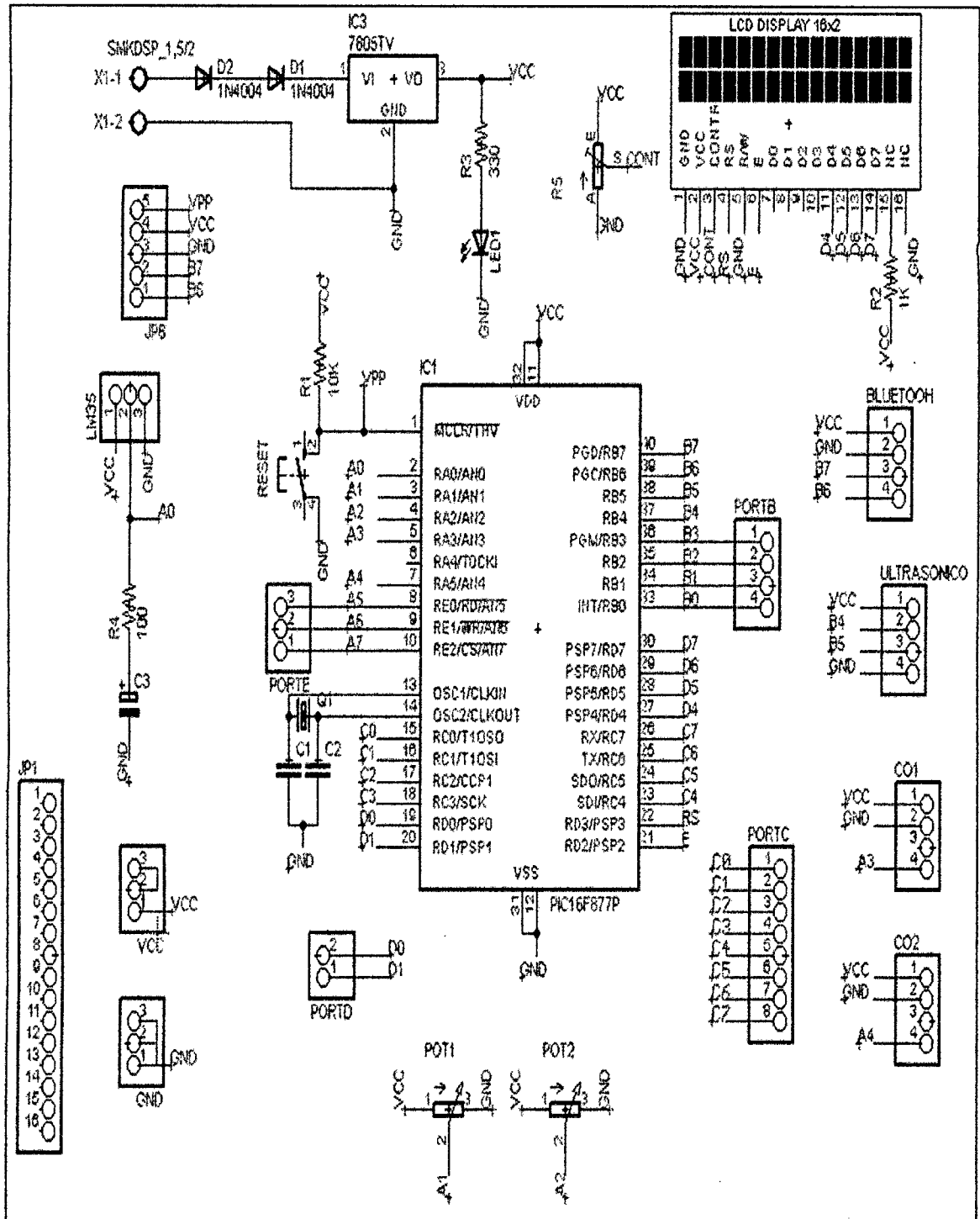


Figura 3.3. Diagrama Electrónico de la tarjeta del Sistema

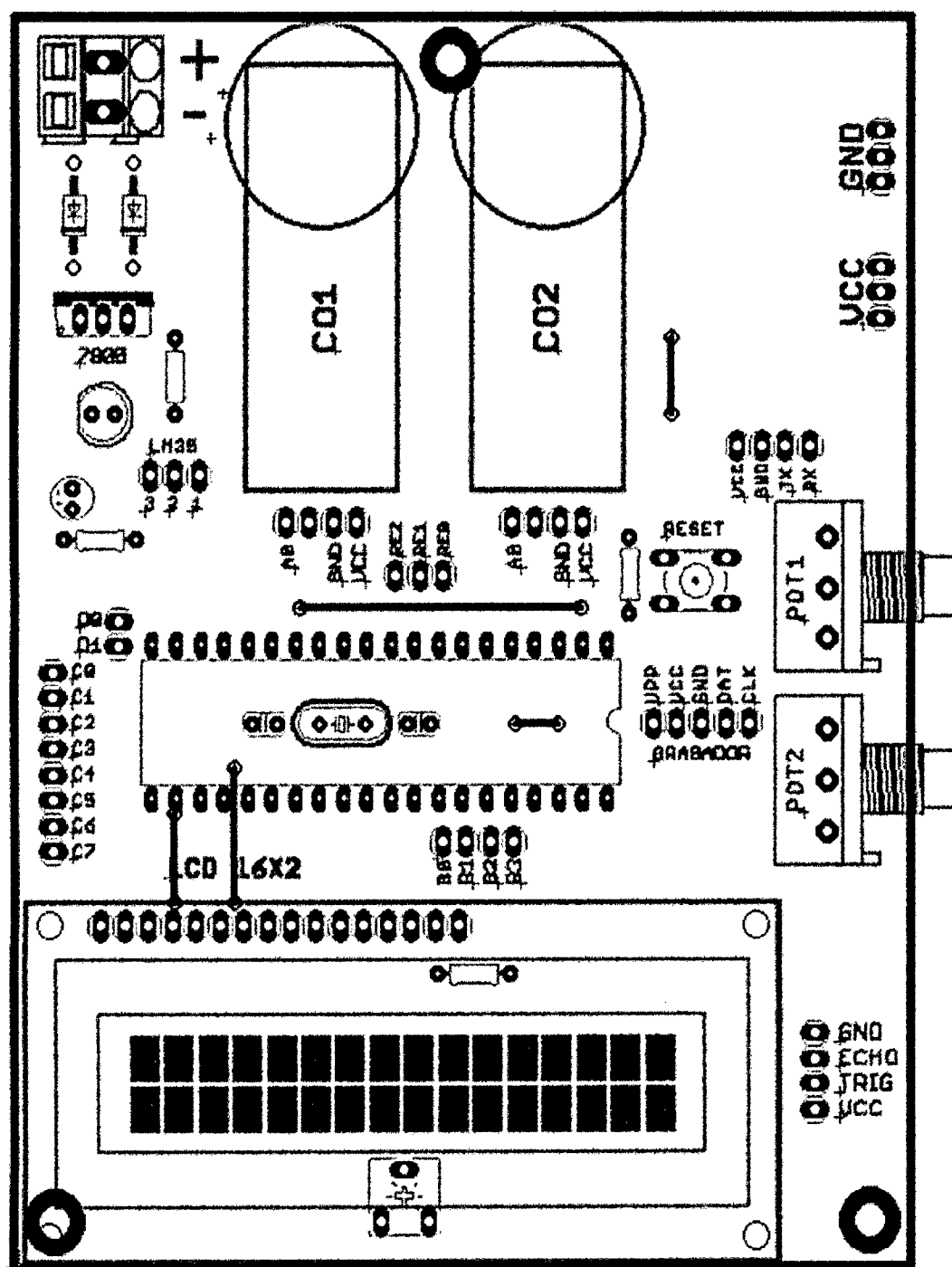


Figura 3.4. Diagrama Electrónico de la tarjeta del Sistema -Serigrafia

3.3. Programación del Microcontrolador

Se procede a describir el código de programa del microcontrolador, en la Figura 3.5. Se muestra el código que corresponde a la configuración del LCD y ADC interno del microcontrolador PIC16F877A

```
*****
DEFINE OSC 4
'DEFINICIONES PARA LCD
DEFINE LCD_DREG PORTD
DEFINE LCD_DBIT 4
DEFINE LCD_RSREG PORTD
DEFINE LCD_RSBIT 3
DEFINE LCD_EREG PORTD
DEFINE LCD_EBIT 2
*****
'CARACTERISTICAS DEL ADC
DEFINE ADC_BITS 8
DEFINE ADC_CLOCK 3
DEFINE ADC_SAMPLEUS 50
TRISA = %11111111 ' Set PORTA to all input
ADCON1 = %00000000 ' Set PORTA analog
*****
```

Figura 3.5 Código de configuración de LC y ADC interno

En la Figura 3.6 se muestran las variables que se utilizaran en el programa.

```
*****
'VARIABLES DEL ADC
ch0 VAR BYTE 'LM35DZ
ch1 VAR BYTE 'POT1
ch2 VAR BYTE 'POT2
ch3 VAR BYTE 'CO1
ch4 VAR BYTE 'CO2
canal0 VAR WORD
canal1 VAR WORD
canal2 VAR WORD
canal3 VAR WORD
canal4 VAR WORD
*****
'VARIABLES PARA MEDIR NIVEL
ECO1 VAR PORTB.5 'PIN DE ECO DEL ULTRASONIDO
TRIGGER VAR PORTB.4 'PIN DE TRIGGER DEL ULTRASONIDO
X VAR WORD ' VALOR EN CM DE LA DISTANCIA
Y VAR WORD ' TIEMPO EN US QUE SE DEMORA EN RETORNAR LA SEÑAL DEL ULTRASON.
XMM VAR WORD ' VALOR EN mm DE LA DISTANCIA
resx VAR WORD ' RESTO DE LA DIVISION DE Y EN TRE 50
d3 VAR WORD ' VALOR DECIMAL EN CM DE LA DISTANCIA
canal5 VAR WORD
*****
```

Figura 3.6 Código de definición de variables

En la Figura 3.7 se muestra el código que lee las variables analógicas y la distancia

```
'PANTALLA INICIAL
LCDOUT $fe, 1
LCDOUT " MEDIDOR TINYWEB"
LCDOUT $fe,$CO," UNP--2014"
PAUSE 1000
*****
inicio:
'LEER CANALES ANALOGICOS
  ADCIN 0, ch0      'LM35DZ
  ch0=ch0*2
  ADCIN 1, ch1      'POT1
  ADCIN 2, ch2      'POT2
  ADCIN 3, ch3      'CO1
  ADCIN 4, ch4      'CO2
*****
  'Leer distancia de sensor de ultrasonido
  HIGH TRIGGER      ' PIN DE TRIGGER DEL ULTRASONIDO SE PONE EN ALTO
  PAUSEUS 10        ' RETARDO DE DE 10US
  LOW TRIGGER       ' PIN DE TRIGGER SE PONE EN BAJO
  ' LUEGO EL SENSOR DE ULTRASONIDO ENVIA SEÑAL DE 40KHZ
  PULSIN ECO1,1,Y ' SE MIDE EL TIEMPO QUE SE DEMORA EN VOLVER LA SEÑAL ENVIADA
  Y=Y*10           ' SE OBTIENE EL TIEMPO EN MICROSEGNDOS
  X =Y/58          ' SE OBTIENE LA DISTANCIA EN cm, x esta en cm,1cm=10mm
  resx=Y//58       ' se obtiene el resto de la división de y en tre 58
  resx=resx*10     ' se multiplica por 10 y
  d3=resx/58       'se obtiene la parte decimal de la distancia. aprox. de un digito
  xmm=X*10         ' SE HACE LA CONVERSION PARA OBTENER DISTANCIA EN MILIMETROS
  xmm=xmm+d3       'SE HACE LA CONVERSION PARA OBTENER DISTANCIA EN MILIMETROS CON F
*****
```

Figura 3.7 Lectura de Variable analógicas y Digital

En la Figura 3.8 se muestra el código para visualizar en LCD el valor de la variables medidas y el envió serial en cadena de datos hacia el modulo Bluetooth.

```

*****
'VIZUALIZAR LOS VALORES DE LA VARIABLES MEDIDAS
LCDOUT $fe, 1
LCDOUT "T=",DEC ch0," P=", DEC ch1, " F=", DEC ch2
LCDOUT $fe,$CO,"C=",DEC ch3," H=", DEC ch4," D=",DEC xtime
PAUSE 100
*****
'ACONDITIONAR LOS VALORES DE LAS VARIABLES PARA QUE CADENA DE DATOS
'SEA DE UN MISMO TAMAÑO SIEMPRE
canal0=ch0+1000
canal1=ch1+1000
canal2=ch2+1000
canal3=ch3+1000
canal4=ch4+1000
canal5=X+1000
*****
' ENVIAR CADENA DE DATOS SERIALMENTE Y POR BLUETOOTH HC06
SEROUT2 PORTB.6,84, {"A",DEC canal0,"B",DEC canal1,"C",DEC canal2,"D",DEC canal3,"E"
PAUSE 50
*****
GOTO inicio ' SALTA A ETIQUETA INICIO PARA INICIAR TODO DE NUEVO
*****

```

Figura 3.8. Visualización de datos en LCD y envío de datos serialmente al módulo HC06

3.4. Software para dispositivo móvil

3.4.1. Especificaciones de la aplicación para el dispositivo movil

La aplicación del dispositivo móvil debe tener las siguientes características:

- Comunicación Bluetooth con la tarjeta de adquisición de datos
- Medir y visualizar en pantalla el valor de minimo 2 variables
- Registrar una Grafica los valores
- Guardar en un archivo jpg la grafica de los Valores Registrados
- Guardar los valores de las variables en la nube (pagina web) con fecha y hora
- Descargar los datos desde la Nube (pagina WEB) mostrarlos y guardarlos en un archivo de texto.

Teniendo en cuenta las especificaciones mencionadas se desarrolla la aplicación la cual tiene la pantalla que se muestra en la Figura 3.5. Y Figura 3.6 Donde se indica las funciones de cada componente.

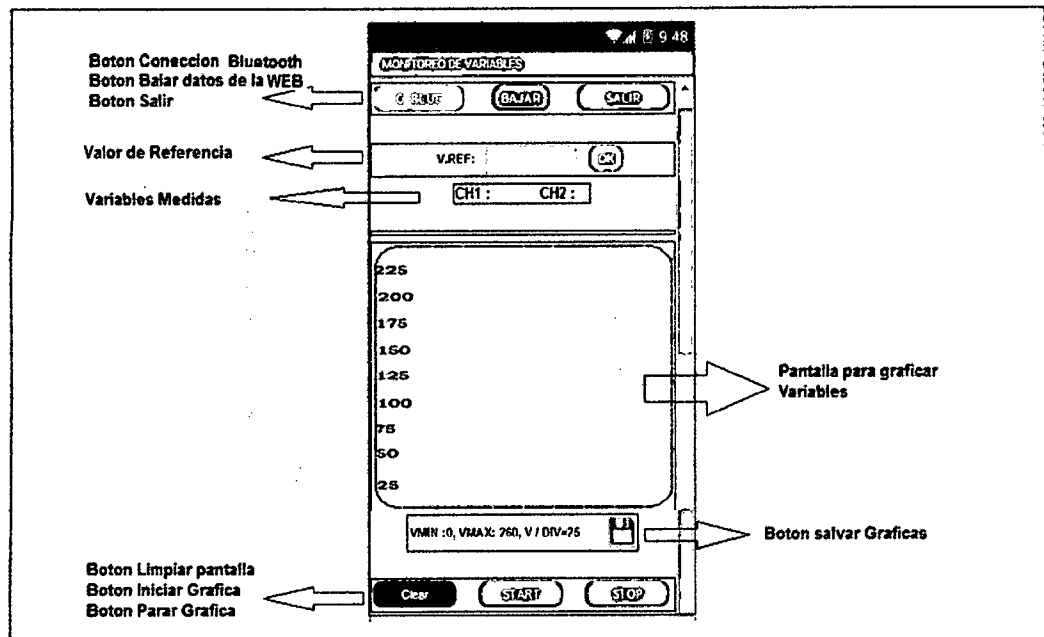


Figura 3.5. Pantalla principal de la aplicación en dispositivo móvil

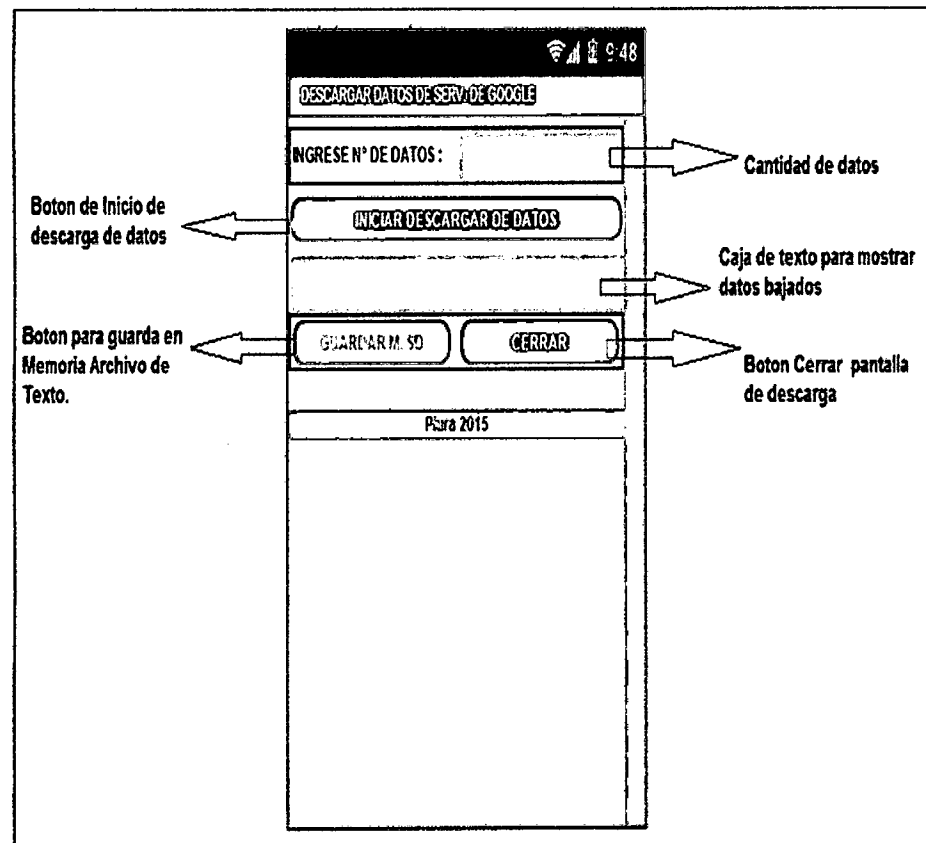


Figura 3.6. Pantalla de descarga de datos

3.4.2. Desarrollo de aplicativo en Appinventor

Para el desarrollo de la aplicación se utilizó el appinventor2, el diseño de la pantalla se muestra en la Figura 3.7 y Figura 3.8. En ella se puede apreciar todos los componentes de la misma.

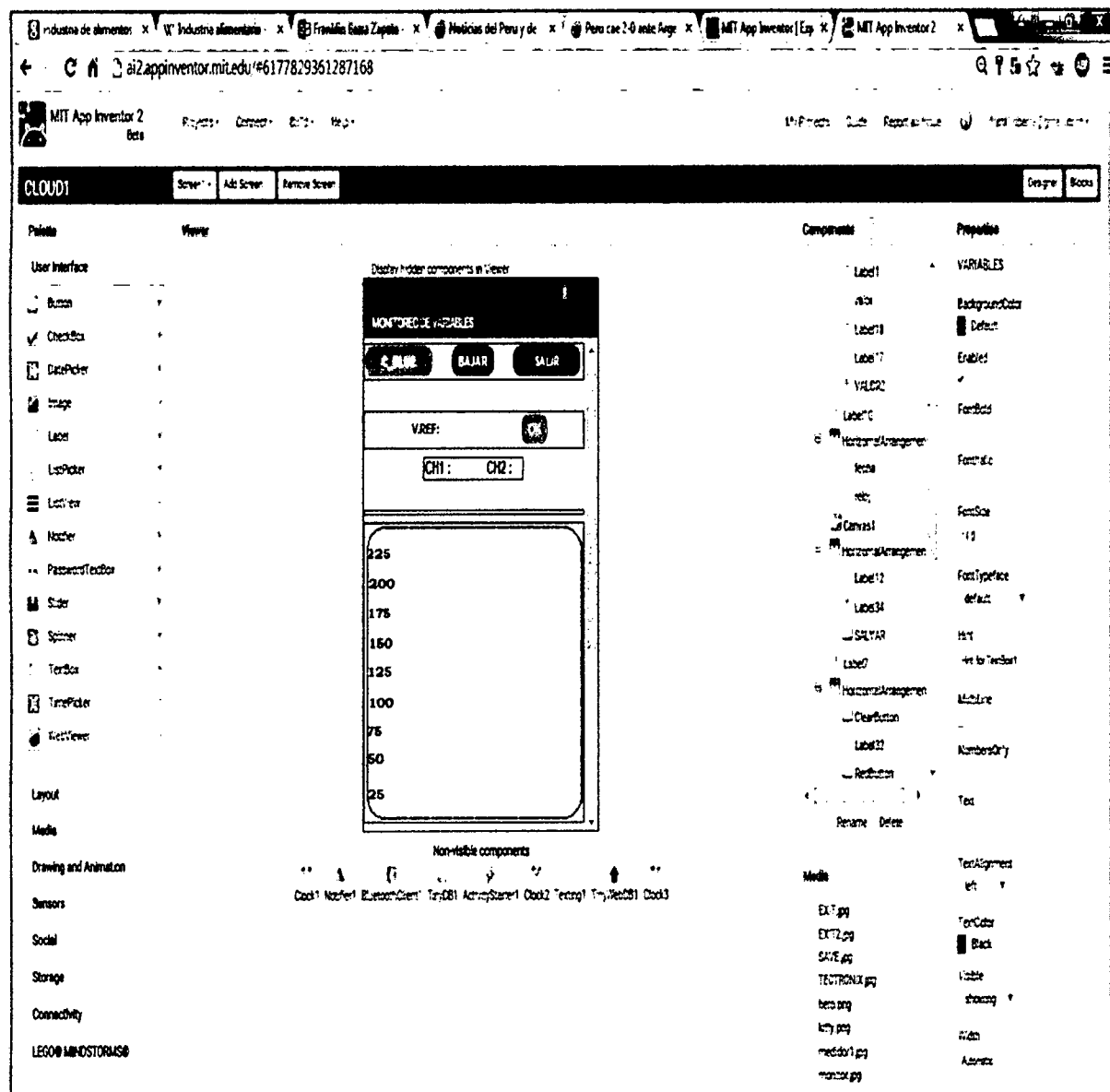


Figura 3.7. Pantalla de Diseño de aplicación en appinventor2 pantalla principal

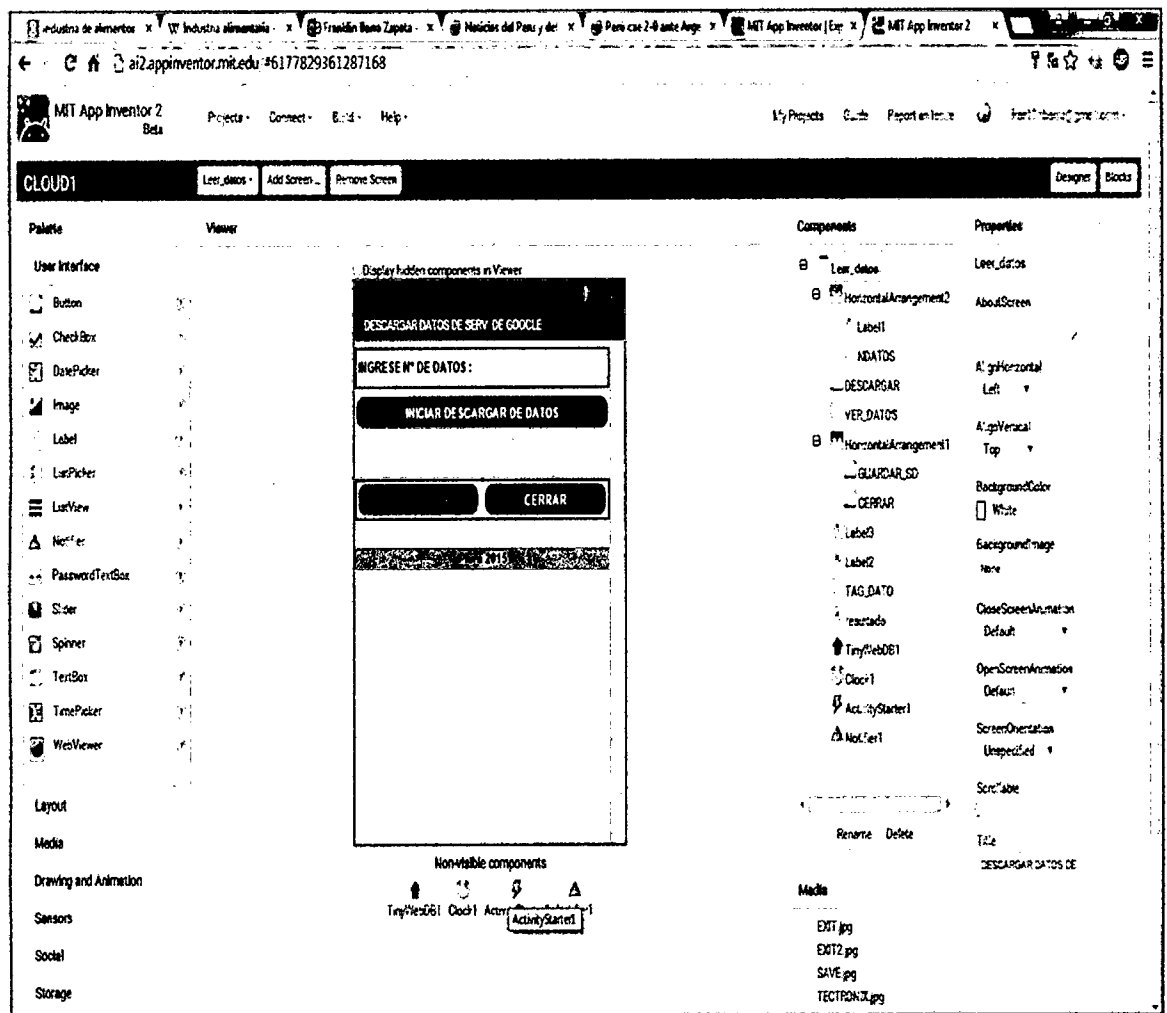


Figura 3.8. Pantalla de Diseño de aplicación en appinventor2 pantalla descarga

3.4.3. Componentes visibles pantalla principal

Los componentes visibles son los que se muestran en la Figura 3.9. y esta compuesto por caja de texto , botones y etiquetas. Los nombres de los componentes se muestran en la Figura3.10

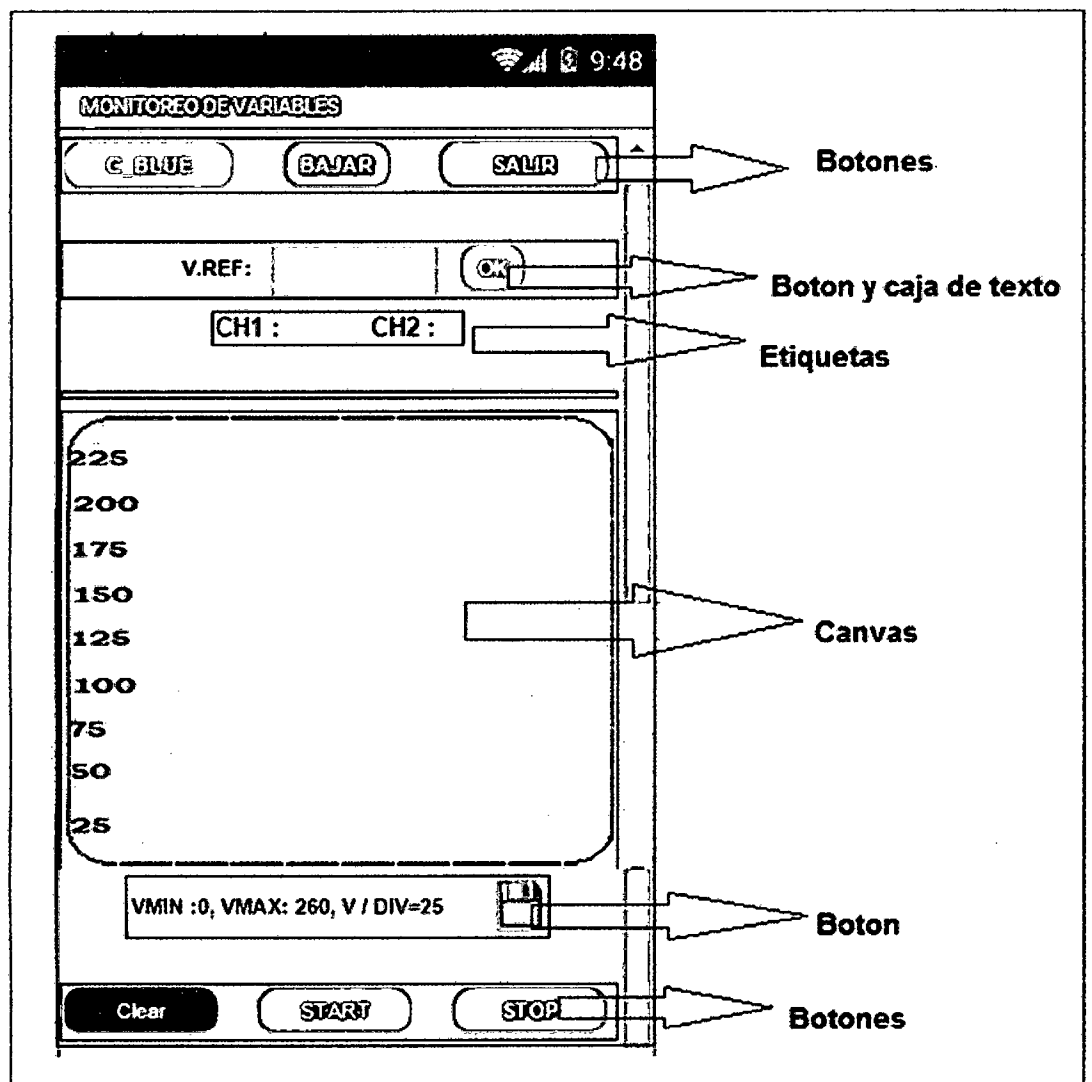


Figura 3.9 Componentes visibles de la pantalla principal

En la Figura 3.10. Se muestran los nombres de los diferentes componentes visibles de la aplicación, estos nombres guardan relación con la función que realizan.

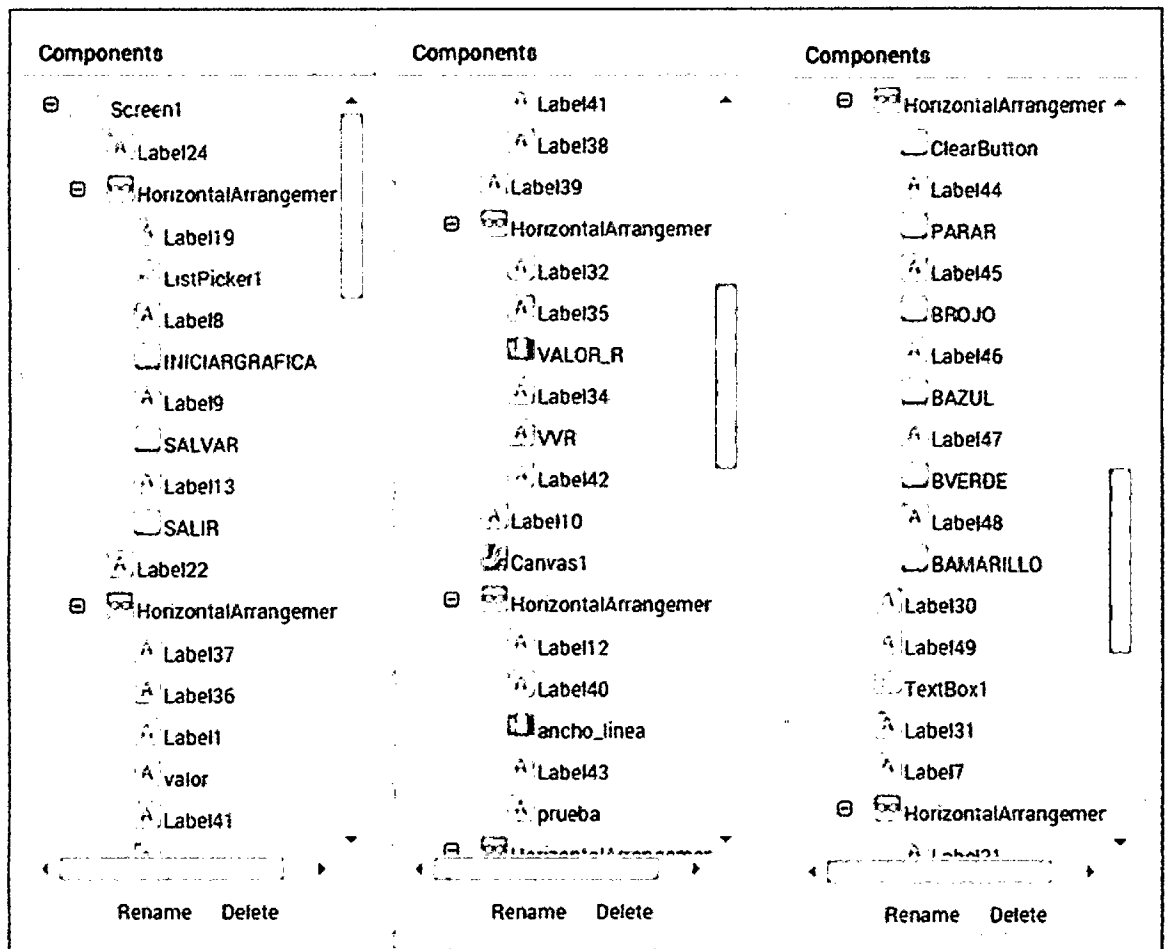


Figura 3.10 Nombres de los diferentes componentes de la pantalla principal

3.4.4. Componentes No Visibles pantalla principal

Estos componentes no son visibles al momento de ejecutar la aplicación pero si son programables por el usuario según requerimientos, los componentes no visibles que se utiliza son los que se muestran en la Figura 3.11.

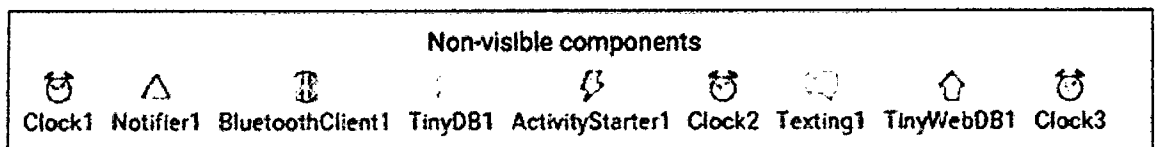


Figura 3.11. Componentes No visibles de pantalla principal

3.4.5. Componentes visibles pantalla de descarga de datos

Los componentes visibles son los que se muestran en la Figura 3.12. y esta compuesto por caja de texto , botones y etiquetas. Los nombres de los componentes se muestran en la Figura3.13

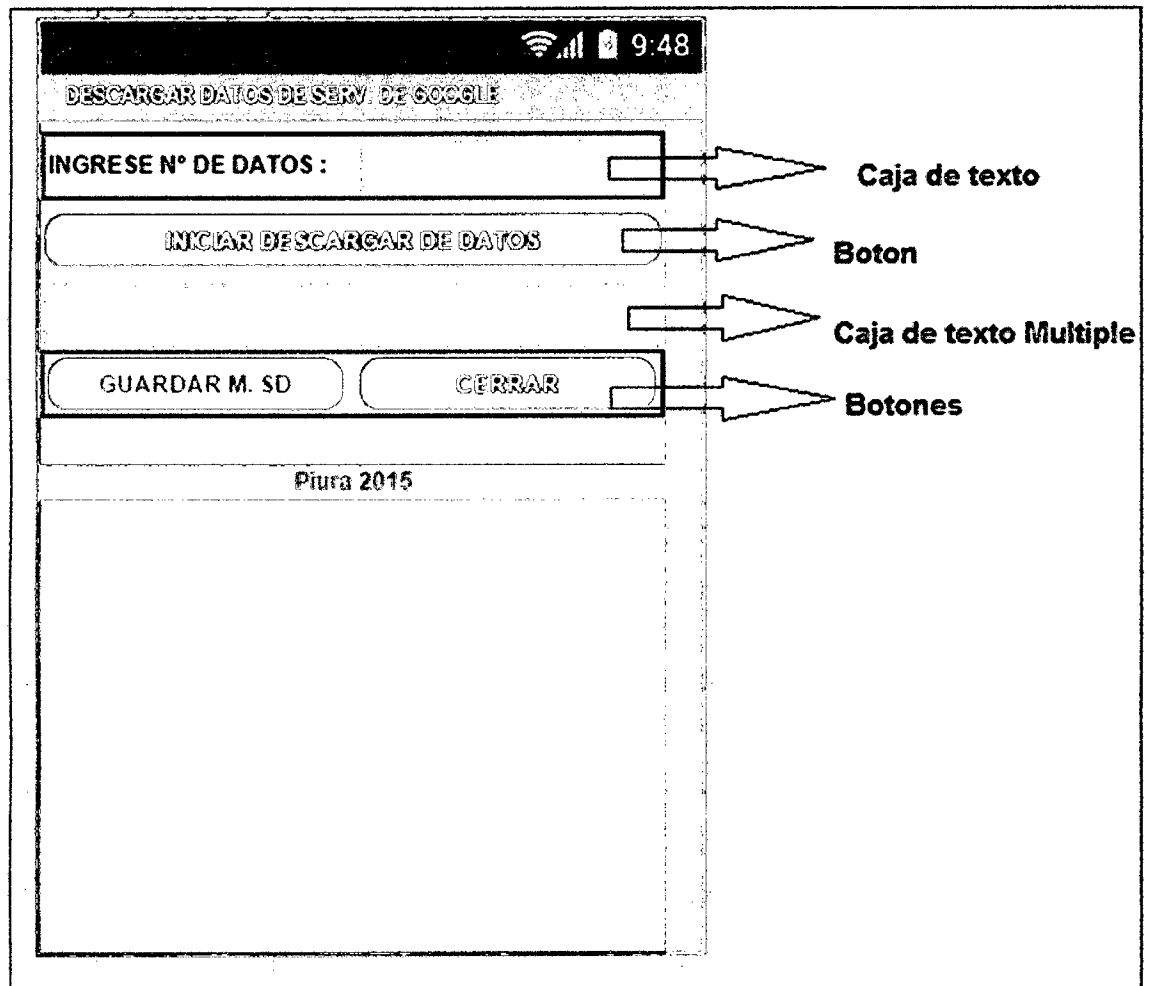


Figura 3.12 Componentes visibles de la pantalla de descarga de datos

En la Figura 3.13. Se muestran los nombres de los diferentes componentes visibles de la aplicación, estos nombres guardan relación con la función que realizan.

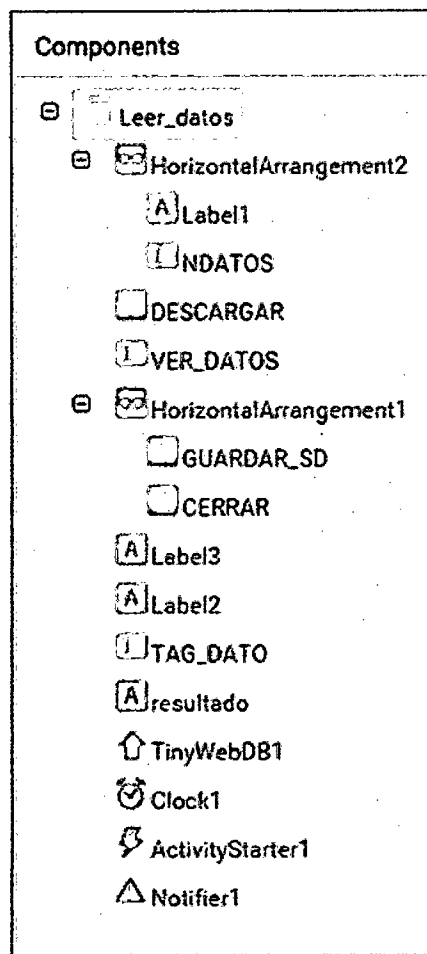


Figura 3.13 Nombres de los diferentes componentes de la pantalla de descarga

3.4.6. Componentes No Visibles pantalla de descarga de datos

Estos componentes no son visibles al momento de ejecutar la aplicación pero si son programables por el usuario según requerimientos, los componentes no visibles que se utiliza son los que se muestran en la Figura 3.14.

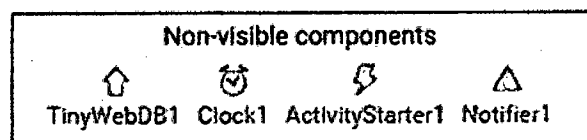


Figura 3.14. Componentes No visibles de pantalla de descarga de datos

3.4.7. Descripción del Programa de pantalla principal

3.4.7.1. Variables

Las variables px1, px2, px3, px4, py1,py2, py3 y py4son utilizadas para graficar línea por línea, uniendo dos puntos (px1, py1; px2,py2; px3, py3; px4, py4). La variable k tomará el valor de 1 si el dispositivo móvil recibe la cabecera de la cadena (A1). La variable VR almacenará el valor de referencia. La variable t=4, es para indicar el inicio de la gráfica en la pantalla (eje x). La variable i es utilizada para la creación del archivo con extensión .jpg. La variable SMS es utilizada para activar el envío del SMS.

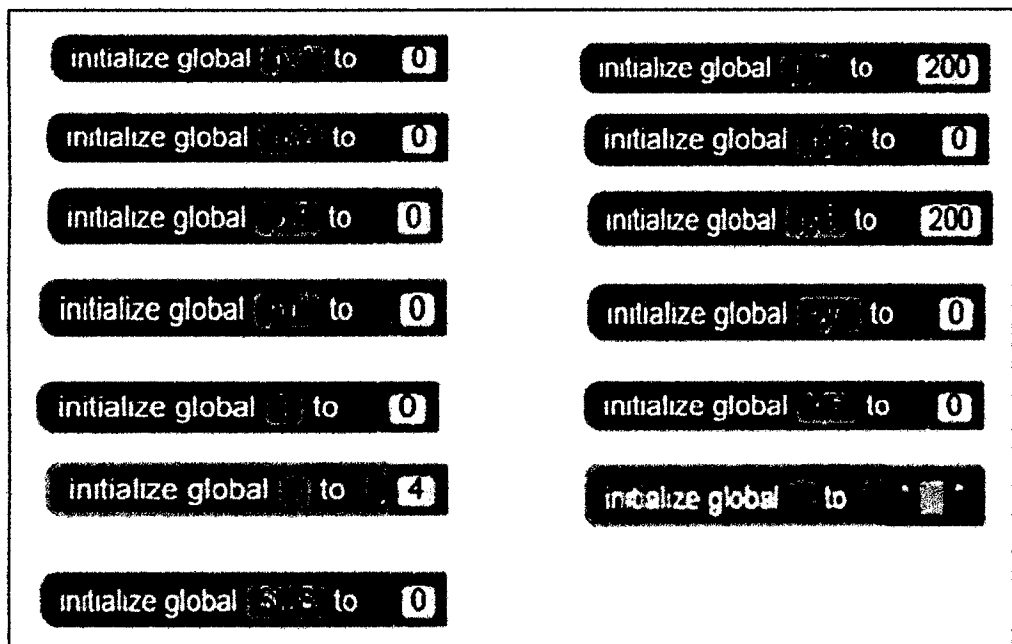


Figura 3.15. Diseño de declaración de variables del Editor de Bloques de App Inventor2

3.4.7.2. BOTONES

Botón ListPicker1- Before picking

Al hacer clic en el botón ListPicker1, el Bluetooth busca al dispositivo sincronizado, si el módulo Bluetooth del Celular después de buscar un registro, y no se dan resultados, se asume que el módulo Bluetooth está y se genera una notificación: “Por Favor Encender Bluetooth”.

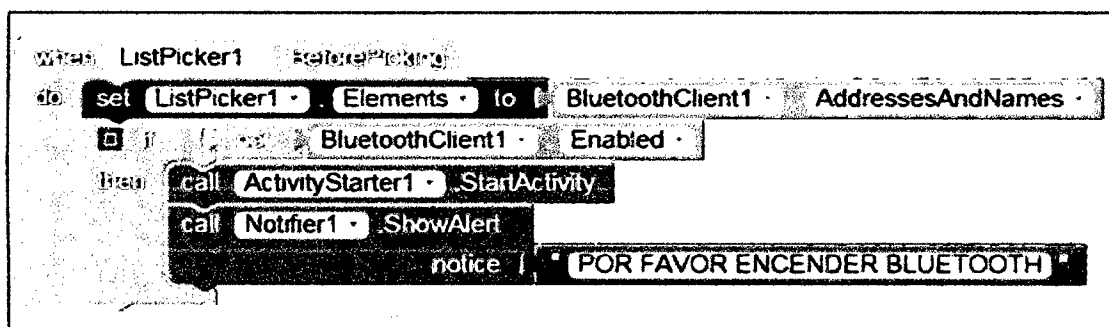


Figura 3.16: ListPicker (Before Picking)

Botón ListPicker1- After Picking

Al hacer click en el Botón ListPicker1 – AfterPicking se nos muestra una lista de los modulos Bluetooth Sincronizados. Se conecta con el Módulo Bluetooth HC-06 que corresponde a la tarjeta de adquisición. Al realizar la conexión este botón se pone de color azul.

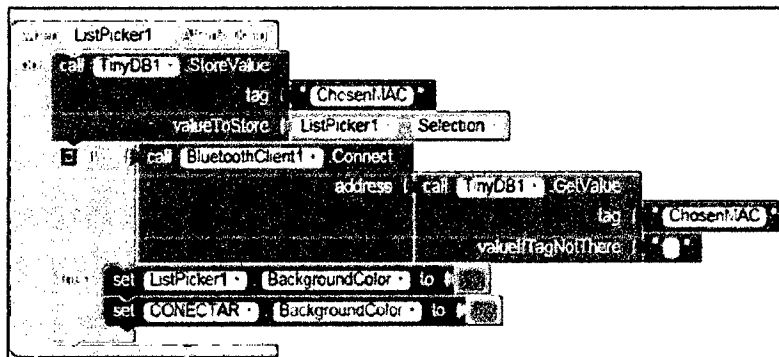


Figura 3.17: Diseño del Botón ListPicker1

Botón SALIR

Al hacer click en el botón SALIR, cortamos comunicación con el Módulo Bluetooth HC -06 y cerramos la aplicación.



Figura3.18: Diseño del Botón SALIR

Botón PARAR

Al hacer click en el botón PARAR, desactivamos el temporizador (Clock1)

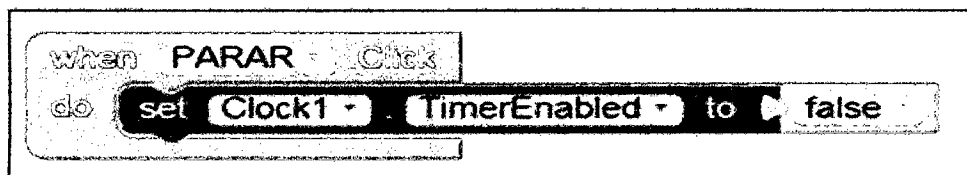


Figura 3.19: Diseño del botón PARAR

Botón SALVAR

Al hacer click en el botón SALVAR, utilizamos el evento notificación para asignar el nombre de la imagen a guardar.

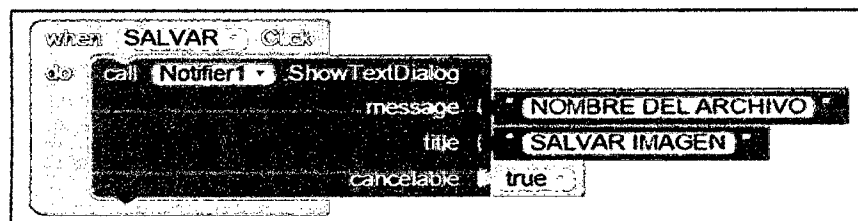


Figura 3.20: Diseño del Botón SALVAR

Botón RedButton

Al hacer click en el botón RedButton activamos el temporizador (clock 1).

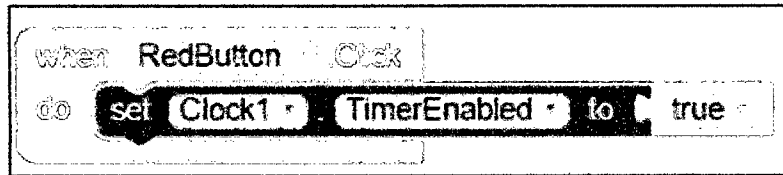


Figura3.21: Diseño del botón RedButton

Botón ClearButton

Al hacer click en el botón ClearButton, asignamos inicio de la gráfica en eje Horizontal ($t=4$), limpiamos la herramienta de animación de la gráfica (canvas) y desactivamos el temporizador (Clock1).

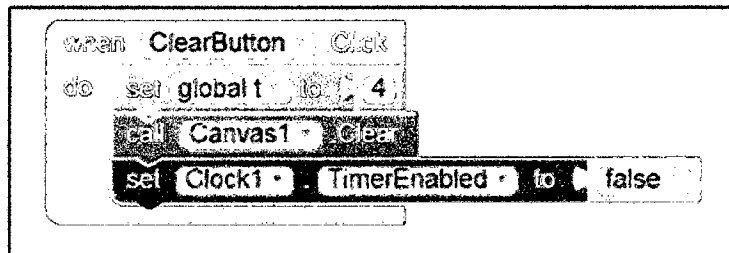


Figura 3.22: Diseño del botón ClearButton

3.4.7.3. DESLIZADOR

Deslizador Silder1

Al variar el deslizador Silder1, éste asigna valores numéricos entre 2-200 para varia Línea de referencia.

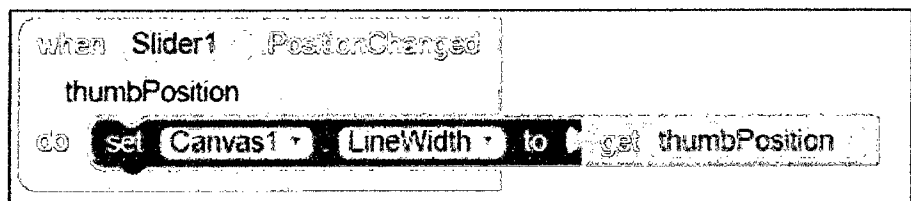


Figura.3.23 Variación de Línea de referencia

3.4.7.4. TEMPORIZADOR

Temporizador Clock1

Se recibe la cadena que envía el Módulo Bluetooth HC-06 de 12 bytes (A1025B1140F13), si el dispositivo móvil recibe la cabecera de la Cadena (A1) entonces k tomará el valor de 1. Luego se compara, si es k es igual a 1 entonces, se extrae un segmento de la cadena desde el byte 2 hasta el byte 4 (1025) obteniendo 1025, a continuación le restamos 1000, obteniendo como resultando el valor del sensor de Temperatura (25), este clock1 guiándose de los 12 bytes que recibe, se obtiene los valores de las variables (temperatura, nivel).

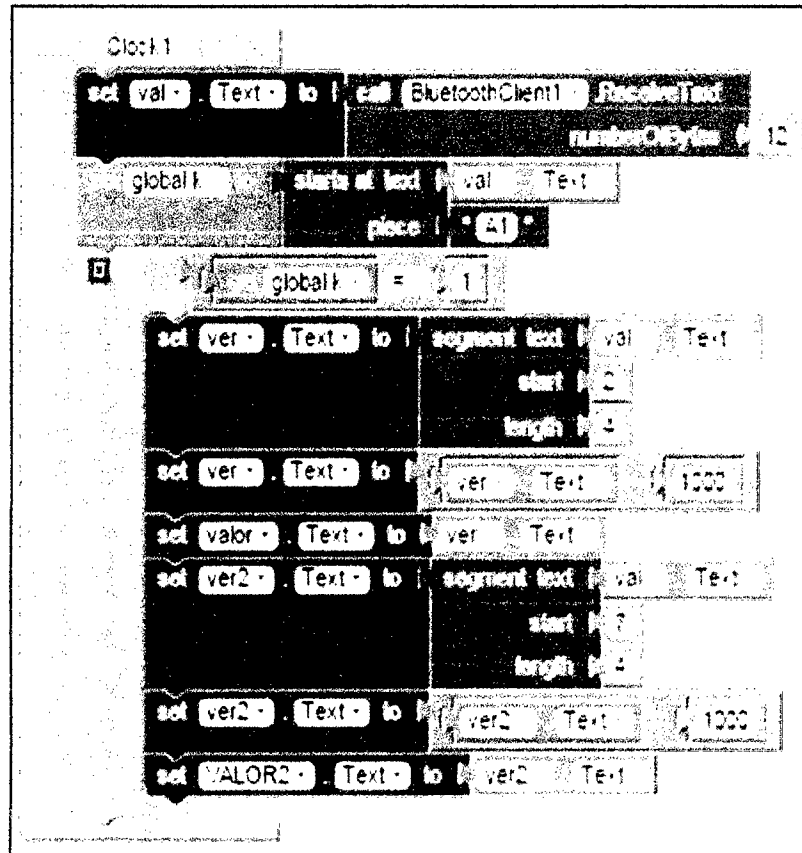


Figura 3.24. Diseño del Clock1 del Editor de Bloques de AppInventor2

Temporizador Clock2

Este temporizador grafica las variables en el Canvas. Px1, Py1, Px2, Py2, Px3, Py3 y Px4, Py4 son las coordenadas o puntos que al unirse forman una recta. El ancho de la pantalla donde se grafica es de los 320 píxeles que al momento de unirse forman una recta de color ROJO para una variable y azul para la otra variable

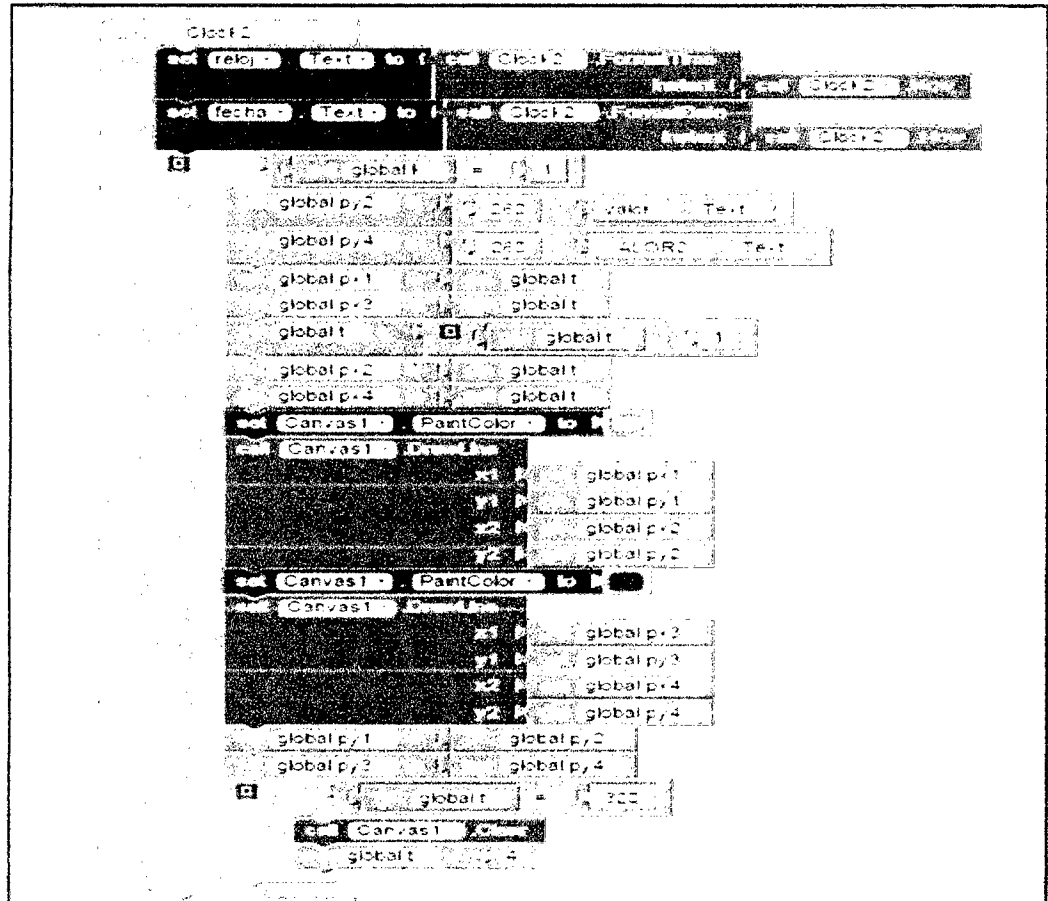


Figura 3.25: Diseño del Clock2

Temporizador Clock3

Este temporizador es el encargado de guardar los datos en la página web de prueba de los servidores de Google

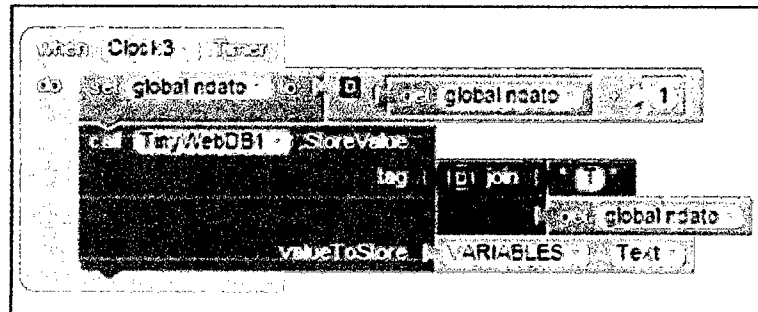


Figura 3.26: Diseño del Clock3

3.4.7.5. NOTIFICACIÓN

Notificación Notifier1

Después de introducido el nombre se guardará la imagen (canvas) con extensiones .jpg, y se visualizará una notificación indicando que la imagen ha sido salvada.

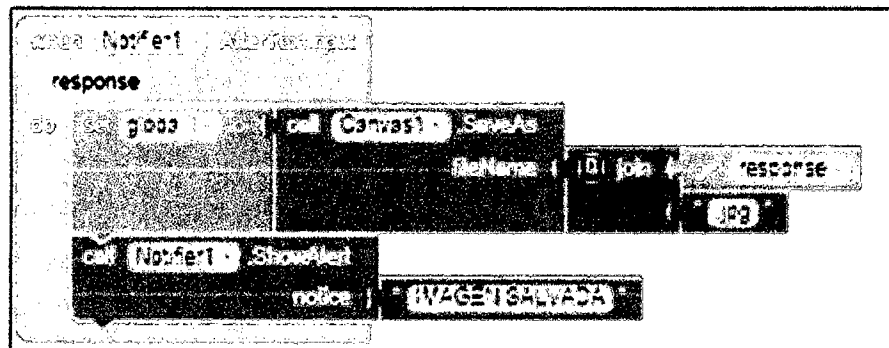


Figura 3.27: Notificación para guardar archivo *.jpg

3.4.7.6. Descargar datos de Servidores de Google

En la Figura pantalla 3.28 se muestra la pantalla para descargar datos de los servidores de google.

Botón Descargar

Al activar este botón se muestra la pantalla de descarga de datos como al que se muestra en la Figura 3.28 y utiliza las variables q se muestran en la Figura 3.29.

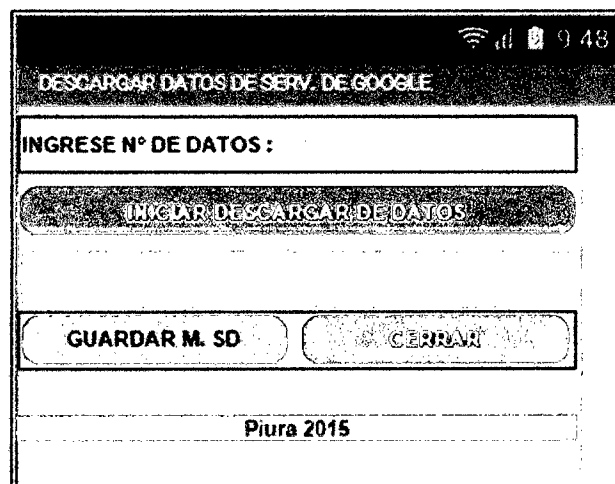


Figura 3.28 Pantalla para descargar datos de servidor de google.

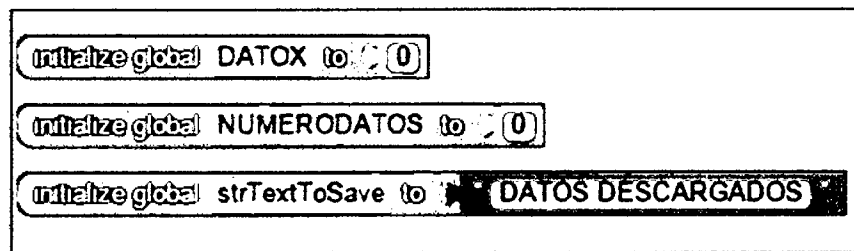


Figura 3.29 Variables Utilizadas para descargar datos de servidores de google.

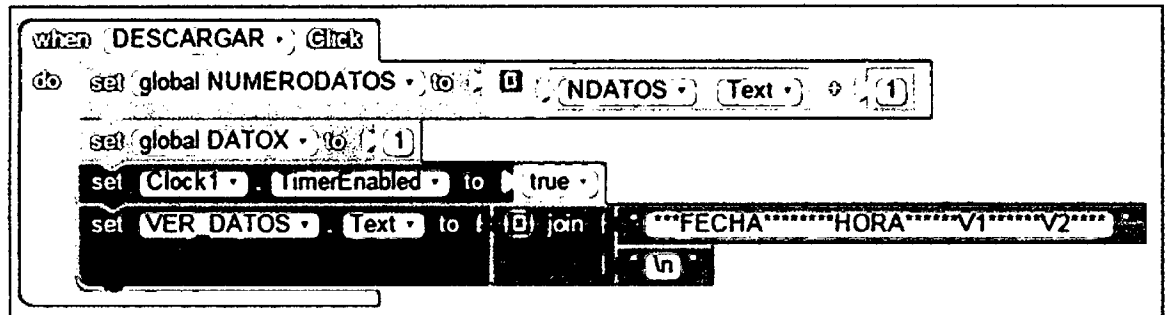


Figura 3.30 Botón de Inicio de descarga de datos

En la Figura 30 se inicia el proceso de descarga de datos para lo cual activa el clock1 (Figura 3.31), el cual llama al componente TinyWebDB1 cada 2 segundos para leer un registro.

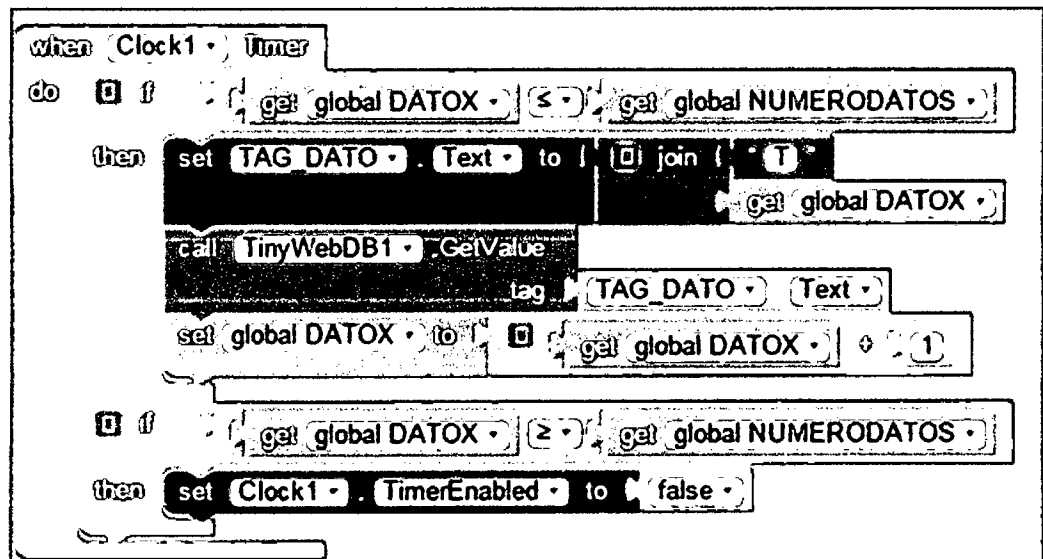


Figura 3.31. Clock1 lectura de un registro del TinywebDB1 cada 2 segundos

Para Visualizar los datos se llama de manera automática al procedimiento GotValue del componente TinyWebDB1 el cual va mostrando en una caja de texto los valores descargados de la página web colgada en los servidores del google. (Figura 3.32)

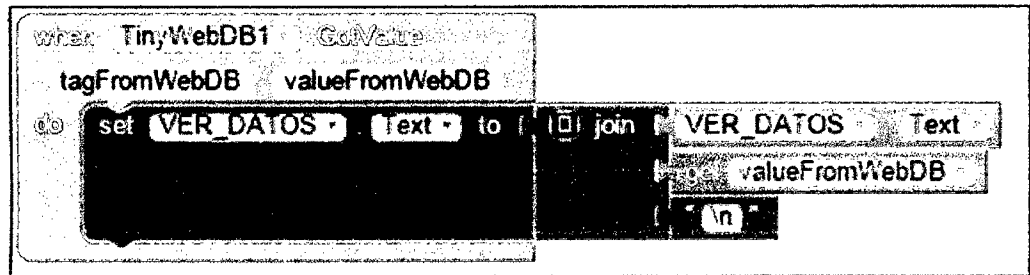


Figura 3.32 Mostrar datos descargados en una caja de Texto.

Botón Cerrar

Este botón se encarga de deshabilitar o cancelar descarga de datos y cierra la pantalla, volviendo a la pantalla principal de la aplicación.(Figura 3.33)

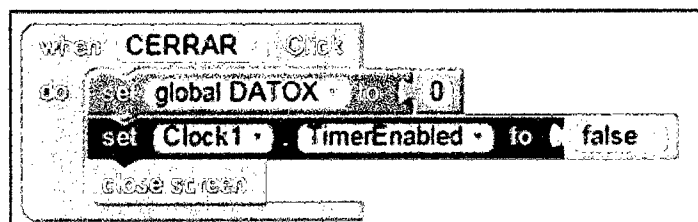


Figura 3.33 Cerrar pantalla de descarga de datos.

Botón Guardar

Este botón se encarga de los datos mostrados en la caja de texto, los guarda en un archivo de texto en el celular o tableta. En la Figura 3.34 y 3.35 se muestra el código correspondiente.

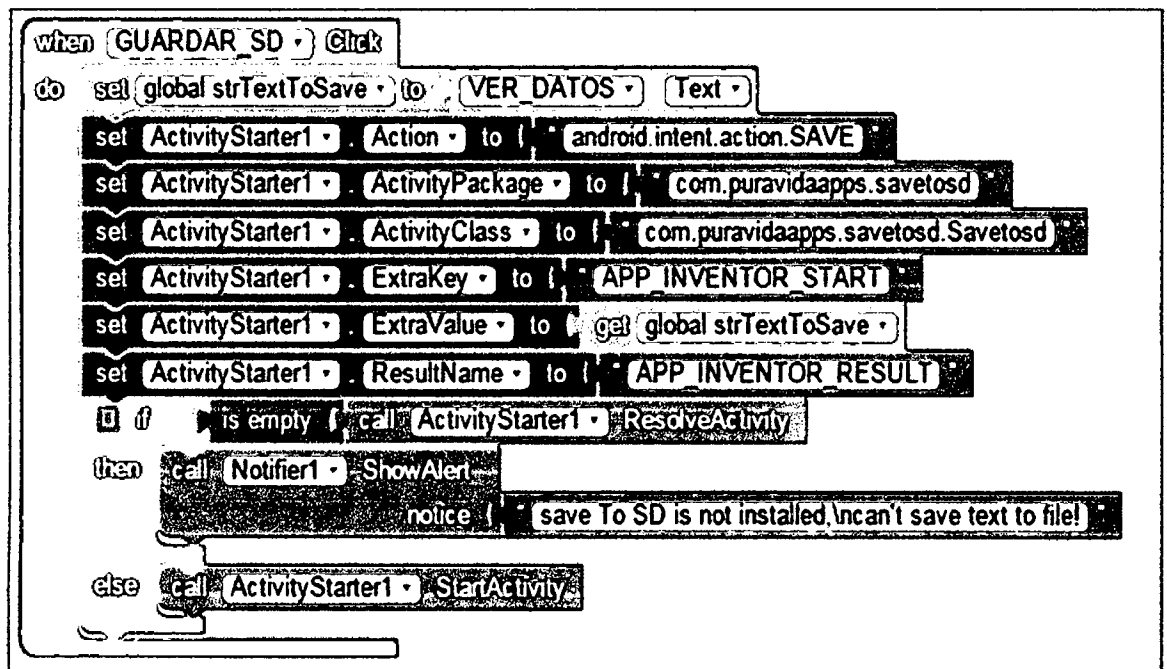


Figura 3.34. Procedimientos para guardar datos en un archivo de texto.

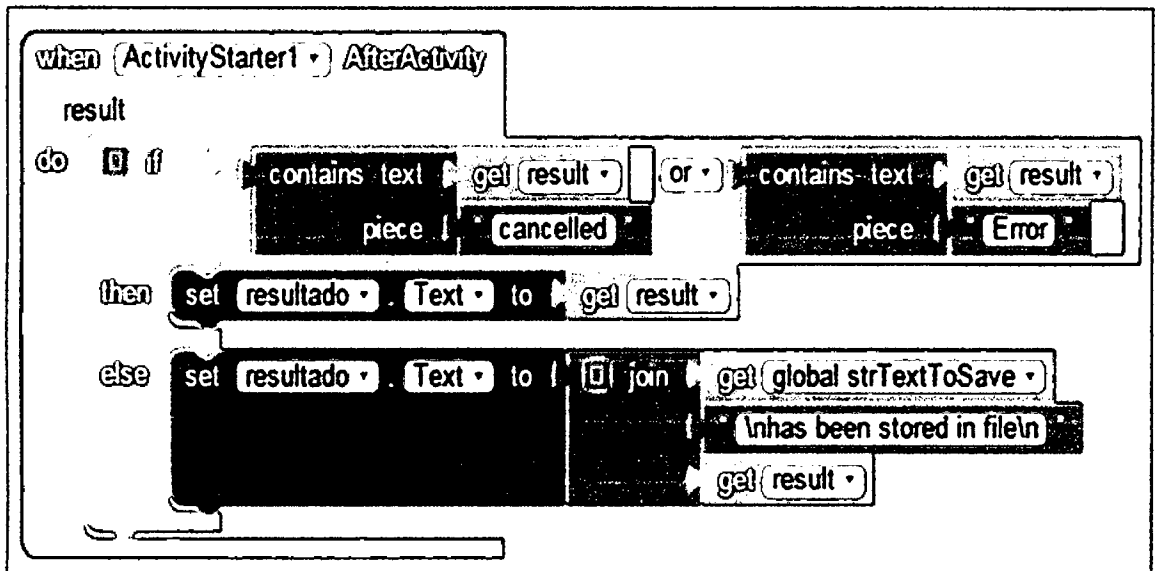


Figura 3.35. Procedimientos para guardar datos en un archivo de texto.

CAPITULO IV

COSTOS DEL PROYECTO

4.1. COSTOS DEL PROYECTO

En este capítulo se describe los costos de implementación del sistema que se plantea.

En los costos que se muestran en la Tabla 4.1, no se tiene en cuenta los costos de Ingeniería como son el costo de diseño y programación, así mismo no se ha considerado los costos de los equipos como Programadores y diversas herramientas necesarias para la implementación del proyecto.

TABLA DE COSTOS 4.1.

COMPONENTES	CANTIDAD	P.UNITARIO	P.TOTAL
Microcontrolador PIC16F877A	1	S/. 40.00	S/. 40.00
Modulo Bluetooth HC06	1	S/. 46.00	S/. 46.00
Modulo sensor de monóxido de carbono CO	1	S/. 50.00	S/. 50.00
LCD Hitech 16x2	1	S/. 25.00	S/. 25.00
Potenciómetro	2	S/. 3.00	S/. 6.00
Sensor LM35DZ	1	S/. 5.00	S/. 5.00
Sensor UltraSonido HCSR04	1	S/. 25.00	S/. 25.00
Componentes básicos varios	1	S/. 30.00	S/. 30.00
Tarjeta de circuito Impreso	1	S/. 100.00	S/. 100.00
Caja para Ckto Electrónico	1	S/. 30.00	S/. 30.00
SUBTOTAL			S/. 357.00
Imprevistos 10%			S/. 35.70
			S/. 392.70

CONCLUSIONES

1. Se logró realizar un prototipo que permite almacenar, información en los servidores de google de los valores de las diferentes variables físicas que se pueden medir en un proceso industrial.
2. Se logró desarrollar un software de interfaz en un celular o tableta con conexión de datos (GPRS) que permite enviar los datos hacia los servidores de google para su almacenamiento.
3. Se logra desarrollar un software que permite descargar los datos almacenados en la nube hacia un archivo de texto el cual luego puede ser procesado usando software común como hojas de cálculo.
4. La computación en la nube es un sistema novedoso, al que cada vez se unen más usuarios y empresas. Tiene muchas ventajas y como suele pasar en los sistemas nuevos también tiene muchos inconvenientes, sobre todo en la seguridad, dependencia del acceso a Internet y de los proveedores de cloud, además la poca madurez de las aplicaciones.
5. Es un sistema que está en sus primeras fases de desarrollo pero cuenta con la ayuda de google y Microsoft y se espera que madure pronto.
6. Las ventajas de cloud computing podemos aprovechar sobre todo para empezar un negocio rápidamente sin grandes inversiones ni instalaciones físicas. Para los sistemas con datos “sensibles” es más recomendable tener la nube privada o híbrida.
7. En el futuro lo que se espera de los proveedores de cloud computing es que mejoren la seguridad y ofrezcan aplicaciones con menos fallos y modificaciones menos frecuentes.
8. Muchas empresas ya tienen sus partes menos sensibles en la nube y en el futuro cercano se espera que la mayoría se migre completamente a la nube.
9. La computación en la nube es una tendencia que se ha posicionado a nivel empresarial como una alternativa para el desarrollo y hosting de aplicaciones web.
10. Existe una gran variedad de proveedores de servicios en la nube, lo que permite a las empresas tener una amplia variedad de ofertas en cuanto a plataformas, capacidades, planes y tarifas.

BIBLIOGRAFÍA

1. Mell, P. and Grance, T. The NIST Definition of Cloud Computing. Gaithersburg: National Institute of Standards and Technology, 2011.
2. Salesforce, <http://www.salesforce.com/es/cloudcomputing/>.
3. Google App Engine - Wikipedia, http://en.wikipedia.org/wiki/Google_App_Engine.
4. Azure Services Platform - Wikipedia, http://en.wikipedia.org/wiki/Azure_Services_Platform.
5. Windows Azure Offers, <http://www.microsoft.com/windowsazure/offers/>.
6. Amazon Web Services - Wikipedia, http://en.wikipedia.org/wiki/Amazon_Web_Services.
7. Engine Yard - Wikipedia, http://en.wikipedia.org/wiki/Engine_Yard.
8. Engine Yard, <http://www.engineyard.com>.
9. Heroku - Wikipedia, <http://en.wikipedia.org/wiki/Heroku>.
10. Software as a Service (SaaS): ¿Qué es?, <http://geeks.ms/blogs/ciin/archive/2007/10/05/software-as-a-service-sas-191-qu-233-es.aspx>.
11. ¿Qué es una Plataforma como Servicio (PaaS)?, <http://www.dosideas.com/noticias/actualidad/504-ique-es-una-plataforma-como-servicio-paas.html>.
12. Infraestructura como Servicio (IaaS) en el Cloud Computing, <http://www.error500.net/software/infraestructura-como-servicio-iaas-cloud-computing>.
13. Reese, George. Cloud Application Architectures. Sebastopol, CA: O'Reilly Media, 2009.
14. Aplicaciones en la Nube, <http://techtextual.com/aplicaciones-en-la-nube/>.

ANEXOS

CODIGO PROGRAMA MICROCONTROLADOR

DEFINE OSC 4

'DEFINICIONES PARA LCD

Define LCD_DREG PORTD

Define LCD_DBIT 4

Define LCD_RSREG PORTD

Define LCD_RSBIT 3

Define LCD_EREG PORTD

Define LCD_EBIT 2

'CARACTERISTICAS DEL ADC

DEFINE ADC_BITS 8

DEFINE ADC_CLOCK 3

DEFINE ADC_SAMPLEUS 50

TRISA = %11111111' Set PORTA to all input

ADCON1 = %00000000 ' Set PORTA analog

'VARIABLES DEL ADC

ch0 var byte 'LM35DZ

ch1 var byte 'POT1

ch2 var byte 'POT2

ch3 var byte 'CO1

ch4 var byte 'CO2

canal0 VAR WORD

canal1 VAR WORD

canal2 VAR WORD

canal3 VAR WORD

canal4 VAR WORD

'VARIABLES PARA MEDIR NIVEL

ECO1 VAR PORTB.5 'PIN DE ECO DEL ULTRASONIDO

TRIGGER VAR PORTB.4 'PIN DE TRIGGER DEL ULTRASONIDO

X VAR WORD ' VALOR EN CM DE LA DISTANCIA

Y VAR WORD ' TIEMPO EN US QUE SE DEMORA EN RETORNAR LA
SEÑAL DEL ULTRASON.

xmm var word ' VALOR EN mm DE LA DISTANCIA

resx var word ' RESTO DE LA DISVIONS DE Y EN TRE 58

d3 var word ' VALOR DECIMAL EN CM DE LA DISTANCIA

canal5 VAR WORD

'PANTALLA INICIAL

LCDOut \$fe, 1

LCDOut " MEDIDOR TINYWEB"

LCDOut \$fe,\$C0," UNP--2014"

PAUSE 1000

inicio:

'LEER CANALES ANALOGICOS

ADCIN 0, ch0 'LM35DZ

CH0=CH0*2

ADCIN 1, ch1 'POT1

ADCIN 2, ch2 'POT2

ADCIN 3, ch3 'CO1

ADCIN 4, ch4 'CO2

'Leer distacia de sensor de ultrasonido

HIGH TRIGGER ' PIN DE TRIGGER DEL ULTRASONIDO SE PONE EN ALTO

PAUSEUS 10 ' RETARDO DE DE 10US

LOW TRIGGER ' PIN DE TRIGGER SE PONE EN BAJO

' LUEGO EL SENSOR DE ULTRASONIDO ENVIA SEÑAL DE 40KHZ

Pulsin ECO1,1,y ' SE MIDE EL TIEMPO QUE SE DEMORA EN VOLVER LA SEÑAL ENVIADA

y=y*10 ' SE OBTIENE EL TIEMPO EN MICROSEGNDOS

x =y/58 ' SE OBTIENE LA DISTANCIA EN cm, x esta en cm,1cm=10mm

resx=y//58 ' se obtiene el resto de la división de y en tre 58

resx=resx*10 ' se multiplica por 10 y

d3=resx/58 'se obtiene la parte decimal de la distancia. aprox. de un digito

xmm=x*10 ' SE HACE LA CONVERSION PARA OBTENER DISTANCIA EN MILIMETROS

xmm=xmm+d3 'SE HACE LA CONVERSION PARA OBTENER DISTANCIA EN MILIMETROS CON PRECISION

'VIZUALIZAR LOS VALORES DE LA VARIABLES MEDIDAS

LCDOut \$fe, 1

LCDOut "T=",DEC CH0," P=", DEC CH1, " F=", DEC CH2

LCDOut \$fe,\$C0,"C=",DEC CH3," H=", DEC CH4," D=",DEC Xmm

PAUSE 100

'ACONDITIONAR LOS VALORES DE LAS VARIABLES PARA QUE CADENA DE DATOS

'SEA DE UN MISMO TAMAÑO SIEMPRE

canal0=ch0+1000

canal1=ch1+1000

canal2=ch2+1000

canal3=ch3+1000

canal4=ch4+1000

canal5=X+1000

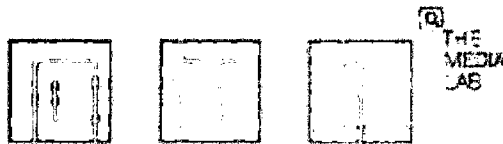
' ENVIAR CADENA DE DATOS SERIALMENTE Y POR BLUETOOTH HC06

SEROUT2 PORTB.6,84, ["A",DEC canal0,"B",dec canal1,"C",dec canal2,"D",dec canal3,"E",dec canal4,"F",dec canal5,10,13]

pause 50

goto Inicio ' SALTA A ETIQUETA INICIO PARA INICIAR TODO DE NUEVO

App Inventor Beginner Tutorials



MIT Center for Mobile Learning

1 Four Simple Tutorials for Getting Started with App Inventor

1.1	TalkToMe: Your first App Inventor app	4
1.2	TalkToMe Part 2: Shaking and User Input	23
1.3	BallBounce: A simple game app	33
1.4	DigitalDoodle: Drawing App	47



MIT App Inventor
appinventor.mit.edu

Four Simple Tutorials for Getting Started with App Inventor



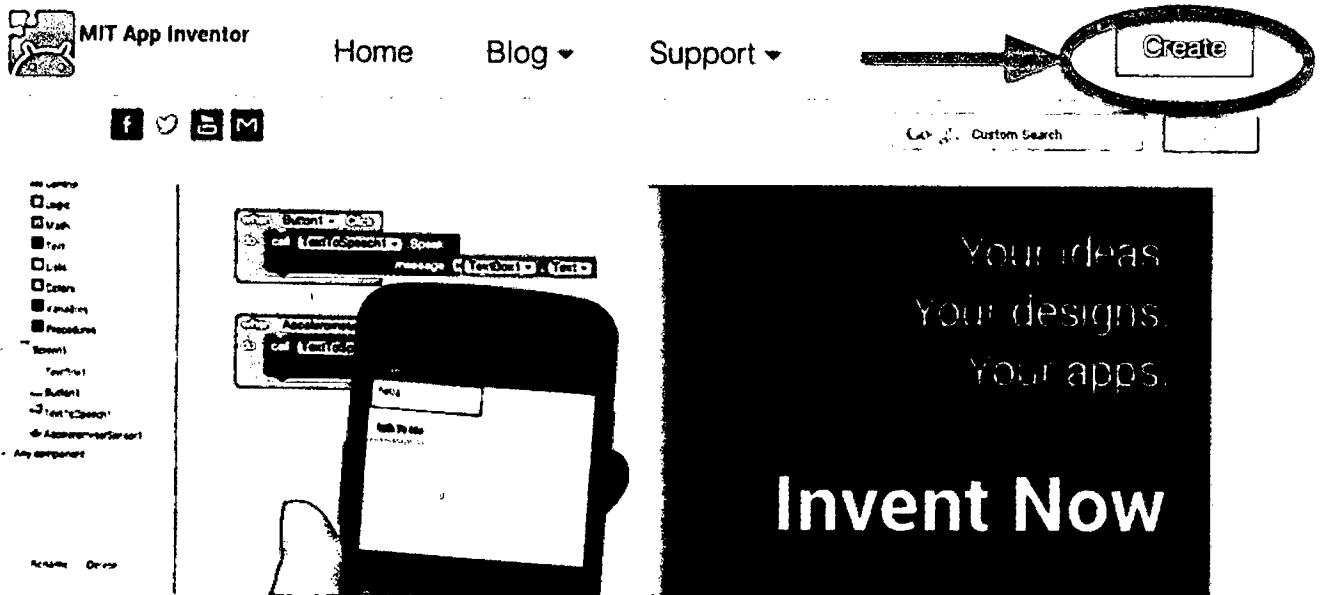
MIT App Inventor
appinventor.mit.edu

TalkToMe: Your first App Inventor app

This step-by-step picture tutorial will guide you through making a talking app.

To get started, go to App Inventor on the web.

Go directly to ai2.appinventor.mit.edu, or click the orange "Create" button from the App Inventor website.



Get Started



Follow these simple steps to build your first app.

Get Started

Create



Design and program your own apps using MIT App Inventor.

Create

Tutorials



Step-by-step guides show you how to build all kinds of apps.

Tutorials



MIT App Inventor
appinventor.mit.edu

Log in to App Inventor with a gmail (or google) user name and password.

Use an existing gmail account or school-based google account to log in to ai2.appinventor.mit.edu
To set up a brand new gmail account, go to accounts.google.com/SignUp

Google

One account. All of Google.

Sign in with your Google Account

appinventorskilz@gmail.com

.....

Sign in

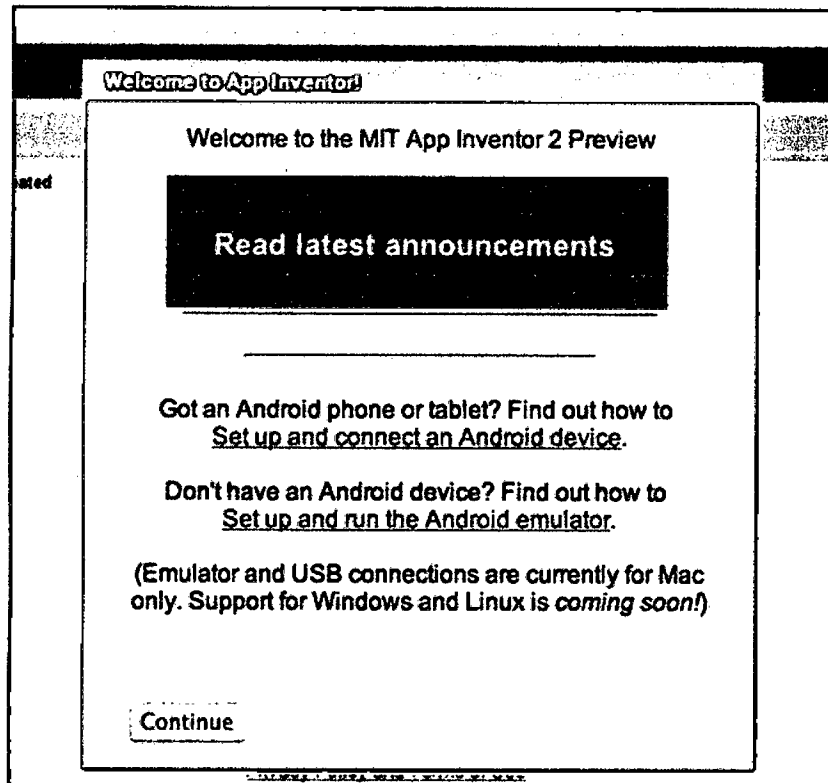
☒ Stay signed in Need help?

[Create an account](#)



MIT App Inventor
appinventor.mit.edu

Click "Continue" to dismiss the splash screen.





MIT App Inventor
appinventor.mit.edu

Start a new project.

MIT App Inventor 2

ai2.appinventor.mit.edu

MIT App Inventor 2

MIT App Inventor 2 Beta Project Connect Build Help My Projects Guide Report an Issue appinventorskilz@gmail.com

New Project ... Delete Project

Projects

Name	Date Created	Date Modified
------	--------------	---------------

Welcome to App Inventor!



You don't have any projects yet. To learn how to use App Inventor, click the "Guide" link at the upper right of the window; or to start your first project, click the "New" button at the upper left of the window.

Happy inventing!

[Privacy Policy and Terms of Use](#)

Name the project "TalkToMe" (no spaces!)

Type in the project name (underscores are allowed, spaces are not) and click OK.

Create new App Inventor project

Project
name:

TalkToMe

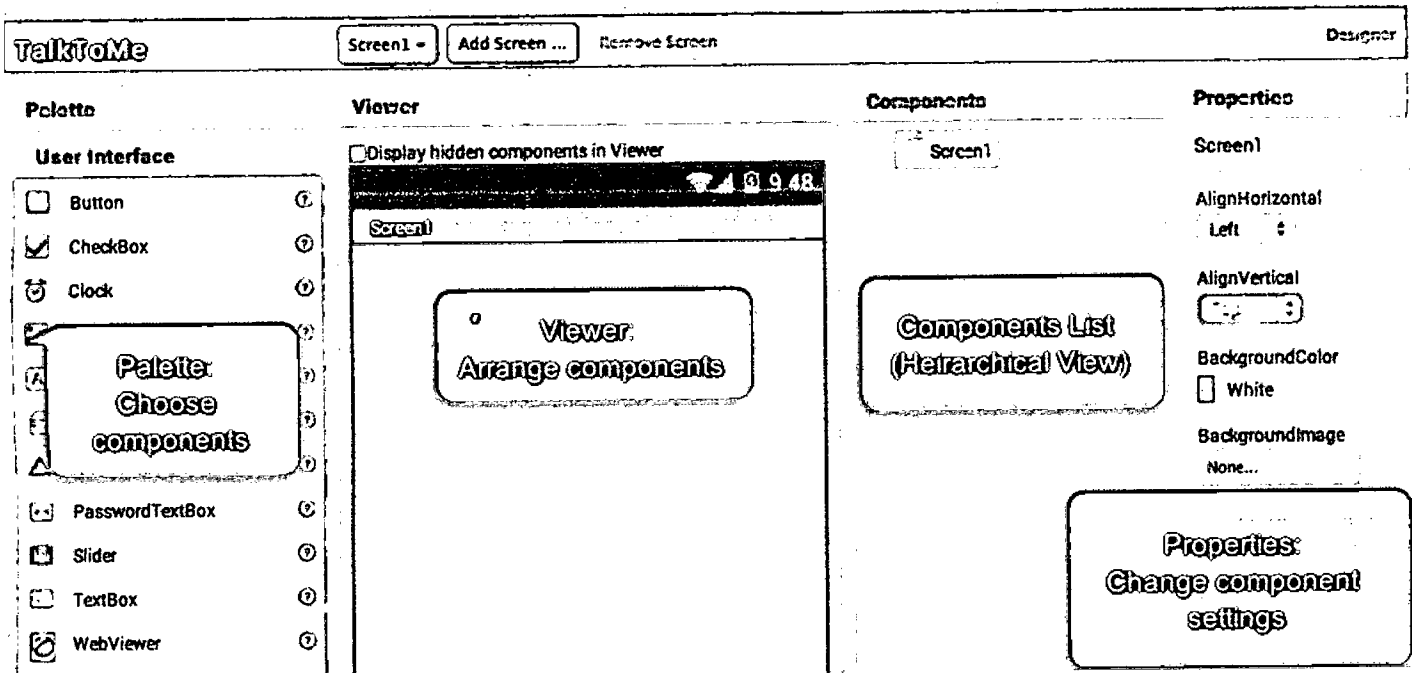
Cancel

OK



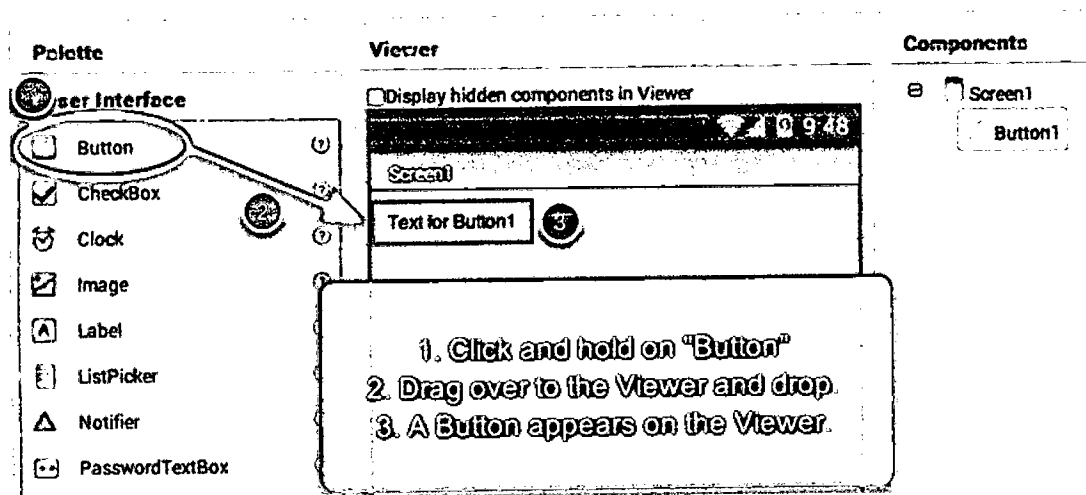
You are now in the Designer, where you lay out the "user interface" of your app.

The Design Window, or simply "Designer" is where you lay out the look and feel of your app, and specify what functionalities it should have. You choose things for the user interface things like Buttons, Images, and Text boxes, and functionalities like Text-to-Speech, Sensors, and GPS.



Add a Button

Our project needs a button. **Click and hold** on the word "Button" in the palette. **Drag** your mouse over to the Viewer. **Drop** the button and a new button will appear on the Viewer.





MIT App Inventor
appinventor.mit.edu

Connect App Inventor to your phone for live testing

One of the neatest things about App Inventor is that you can see and test your app while you're building it, on a connected device. If you have an **Android phone or tablet**, follow the steps below. *If you do not have a device*, then follow the instructions for setting up the on-screen emulator (opens a new page) and then come back to this tutorial once you've gotten the emulator connected to App Inventor.



MIT App Inventor 2
Beta

Project ▾

Connect ▾

Build ▾

Help ▾

TalkToMe

Screen1 ▾

AI Companion

Emulator

USB

Reset Connection

Palette

Viewer

User Interface

☐ Button

☐ Display hidden

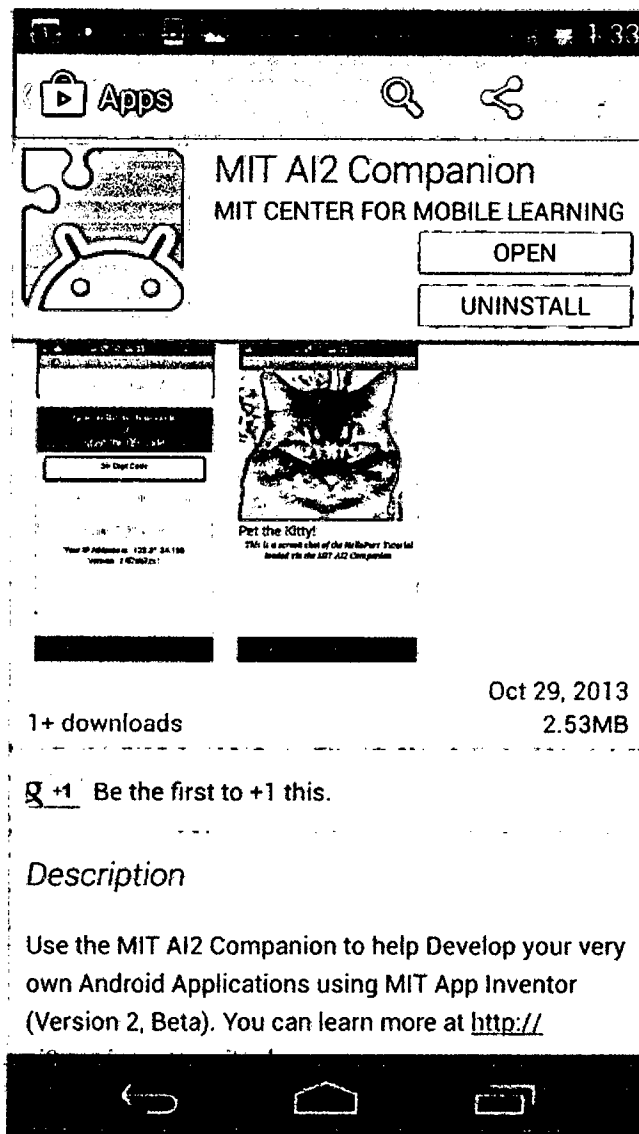
Screen1



MIT App Inventor
appinventor.mit.edu

Get the MIT AI2 Companion from the Play Store and install it on your phone or tablet.

The preferred method for getting the AI2 Companion App is to download the app from the Play Store by searching for "MIT AI2 Companion".





MIT App Inventor
appinventor.mit.edu

To download the AI2 Companion App to your device directly (SKIP THIS STEP IF YOU already got the app from Play Store)

If for some reason you can not connect to the Google Play store, you can download the AI2 Companion as described here.

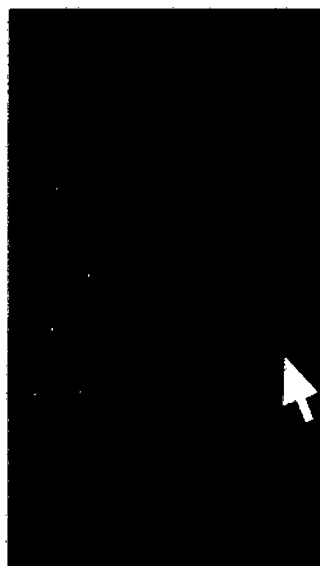
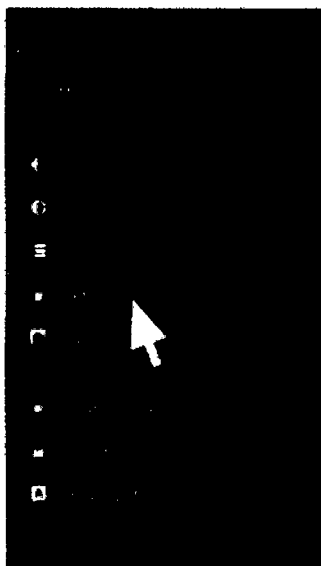
First, you will need to go into your phone's settings (#1), choose "Security", then scroll down to allow "Unknown Sources", which allows apps that are not from the Play Store to be installed on the phone.

Second, do one of the following:

A) **Scan the QR code above (#2)**

or

B) Click the "Need help finding..." link and you'll be taken to the download page. From there you can download the MITAI2Companion.apk file to your computer and then move it over to your device to install it.



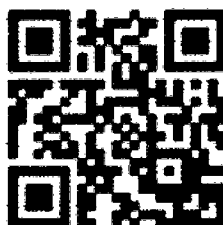
SKIP THIS STEP if you already got the AI2 Companion from the Play Store

1

1. Open your phone's settings and click "Security".
2. CHECK the box for "Unknown sources".

2

Scan to download MIT AI2 Companion directly to phone



If you need help...

Connect to Companion

Launch the MIT AI2 Companion on your device and then scan the barcode or type in the code to connect for live testing of your app. Need help finding the Companion App?



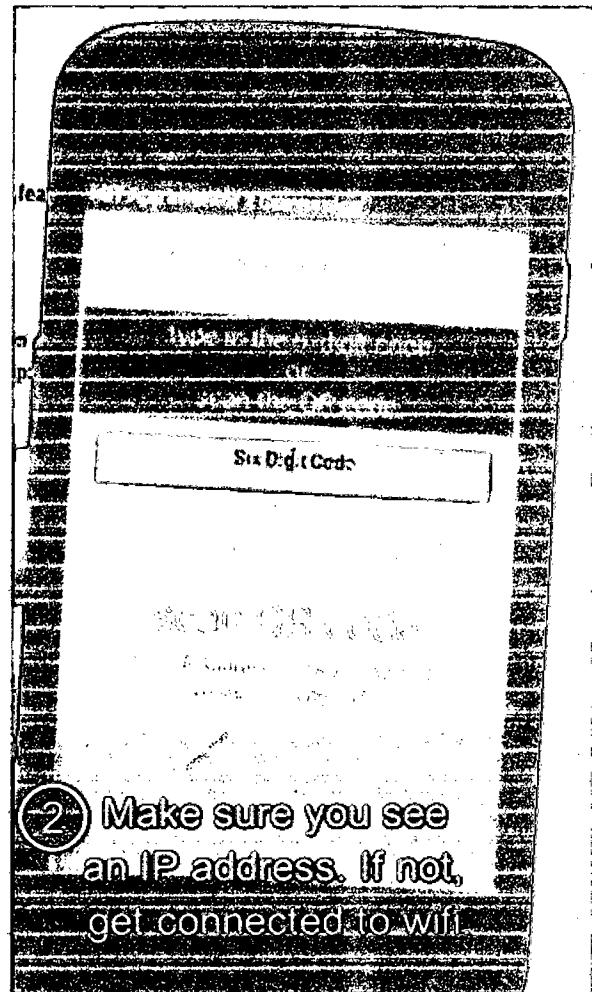
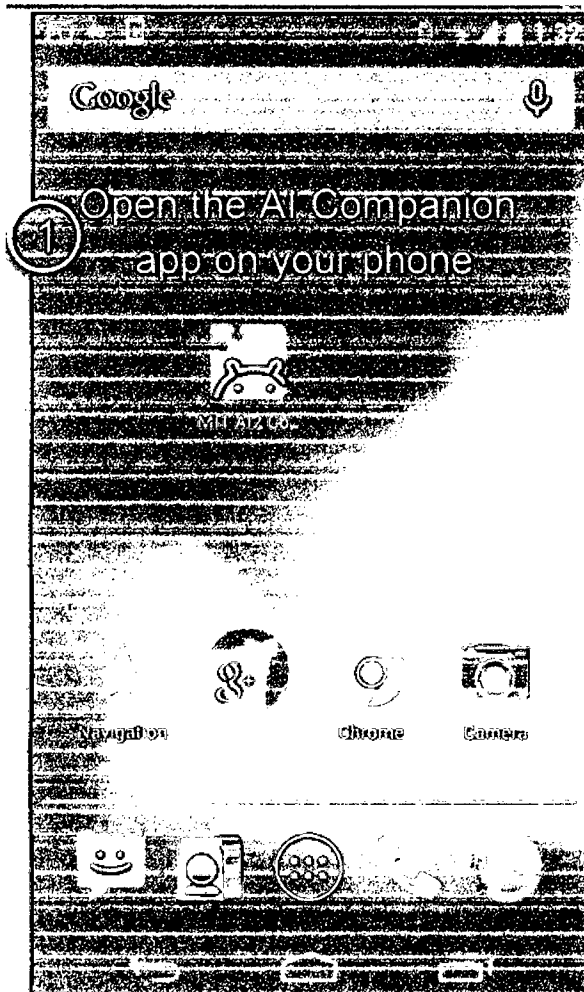
Your code is:



MIT App Inventor
appinventor.mit.edu

Start the AICompanion on your device

On your phone or tablet, click the icon for the MIT AI Companion to start the app. NOTE: Your **phone and computer must both be on the same wireless network**. Make sure your phone's wifi is on and that you are connected to the local wireless network. If you can not connect over wifi, go to the Setup Instructions on the App Inventor Website to find out how to connect with a USB cable.





MIT App Inventor
appinventor.mit.edu

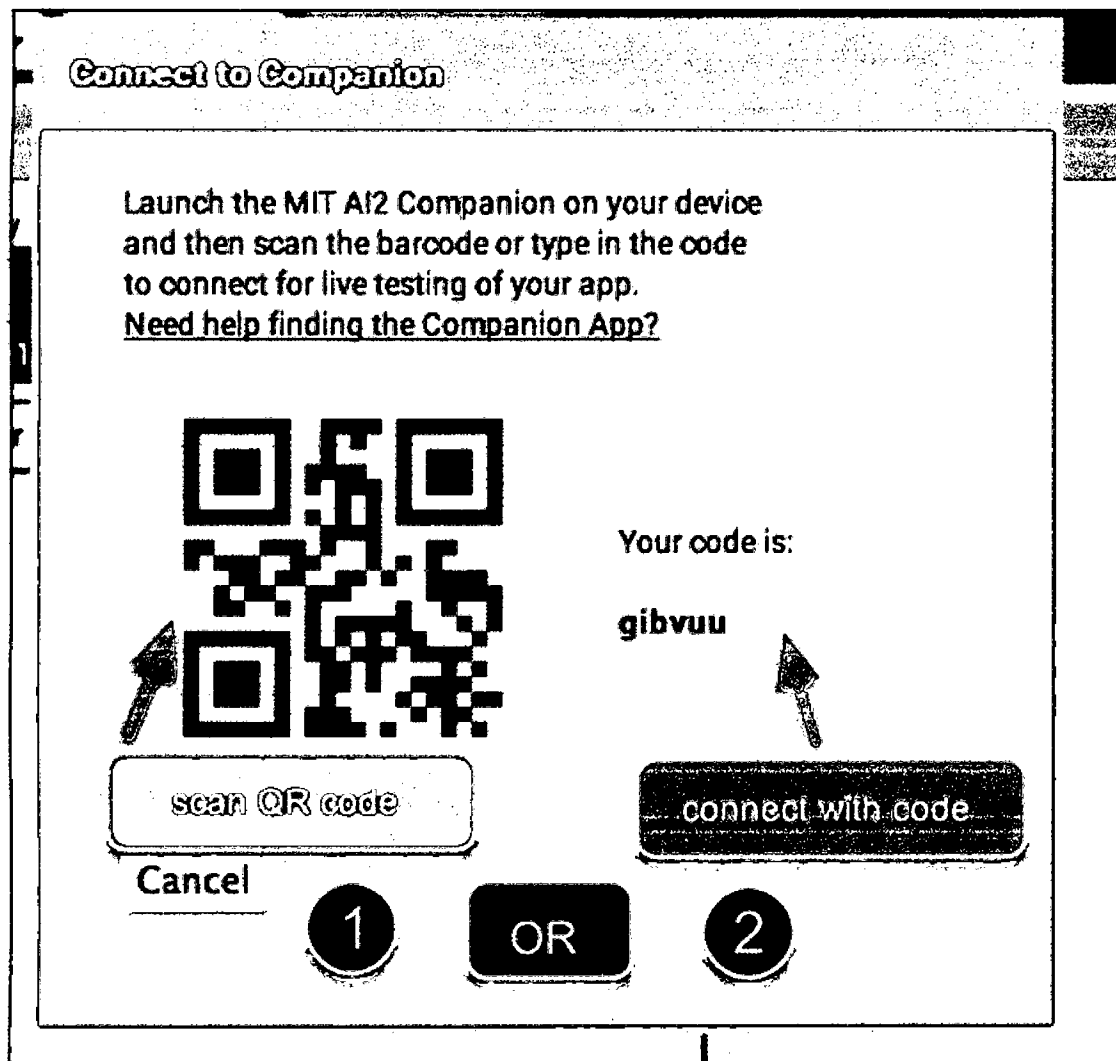
Get the Connection Code from App Inventor and scan or type it into your Companion app

On the Connect menu, choose "AI Companion". You can connect by:

1 - Scanning the QR code by clicking "Scan QR code" (#1).

or

2 - Typing the code into the text window and click "Connect with code" (#2).

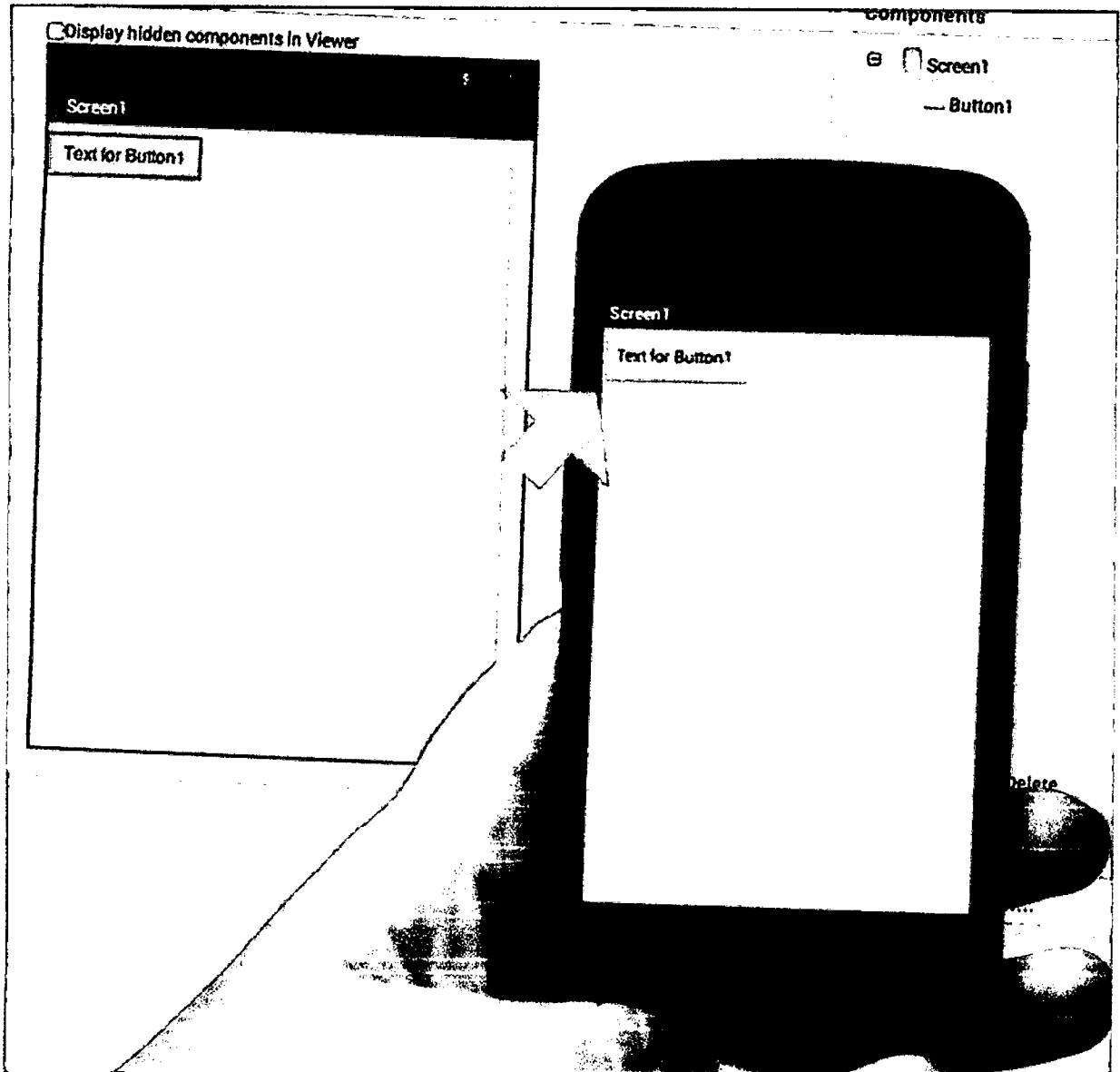




MIT App Inventor
appinventor.mit.edu

See your app on the connected device

You will know that your connection is successful when you see your app on the connected device. So far our app only has a button, so that is what you will see. As you add more to the project, you will see your app change on your phone.





Change the Text on the Button

On the properties pane, change the text for the Button. Select the text "Text for Button 1", delete it and type in "Talk To Me". Notice that the text on your app's button changes right away.

The screenshot displays the MIT App Inventor web interface with three main panels: Viewer, Components, and Properties.

- Viewer Panel:** Shows a mobile app preview. At the top, a black bar contains the text "Screen1". Below it, a white button with a black border contains the text "Talk To Me". A context menu is open over the button, showing options: "Delete", "File ...", "Rename", and "Delete".
- Components Panel:** A tree view showing the app's structure. It includes "Screen1" and a sub-component "Button1".
- Properties Panel:** Lists the properties for the selected "Button1". The properties include: "BackgroundColor" (Default), "Enabled" (checked), "FontBold" (unchecked), "FontItalic" (unchecked), "FontSize" (14.0), "FontTypeface" (default), "Image" (None...), "Shape" (default), "ShowFeedback" (checked), and "Text" (Talk To Me). The "Text" property is highlighted with a red box.



Add a Text-to-Speech component to your app

Go to the Media drawer and drag out a TextToSpeech component. Drop it onto the Viewer. Notice that it drops down under "Non-visible components" because it is not something that will show up on the app's user interface. It's more like a tool that is available to the app.

The screenshot displays the MIT App Inventor web interface with four main panels: Palette, Viewer, Components, and Properties.

- Palette:** The 'Media' category is circled. An arrow points from the 'TextToSpeech' component in the Media list to the Viewer.
- Viewer:** The 'Screen1' area contains a text label 'Talk To Me'. Below the screen, in the 'Non-visible components' area, a 'TextToSpeech1' component has been added. A text box in the center of the screen reads: 'Drop here. Component will automatically show up in Non-visible components area below'.
- Components:** The 'Screen1' component is expanded, showing a list of sub-components: 'Button1' and 'TextToSpeech1'.
- Properties:** The 'TextToSpeech1' component is selected, showing properties: 'TextToS', 'Country', and 'Language'.

At the bottom right, there are buttons for 'Rename' and 'Delete', and a section for 'Media' with an 'Upload File ...' button.

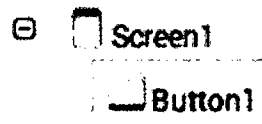


Switch over to the Blocks Editor

It's time to tell your app what to do! Click "Blocks" to move over to the Blocks Editor. Think of the Designer and Blocks buttons like tabs -- you use them to move back and forth between the two areas of App Inventor.

[My Projects](#)[Guide](#)[Report an Issue](#)[appinventorskilz@gmail.com](#) ▼**Designer****Blocks**

Components



Properties

Button1**BackgroundColor**☒ Default**Enabled****FontBold****FontItalic**



The Blocks Editor

The Blocks Editor is where you program the behavior of your app. There are Built-in blocks that handle things like math, logic, and text. Below that are the blocks that go with each of the components in your app. *In order to get the blocks for a certain component to show up in the Blocks Editor, you first have to add that component to your app through the Designer.*



Make a button click event

Click on the Button1 drawer. Click and hold the **when Button1.Click do** block. Drag it over to the workspace and drop it there. This is the block that will handle what happens when the button on your app is clicked. It is called an "Event Handler".



MIT App Inventor 2
Beta

Project ▾ Connect ▾ Build ▾ Help ▾

My Projects Guide Rep

TalkToMe

Screen1 ▾

Add Screen ...

Remove Screen

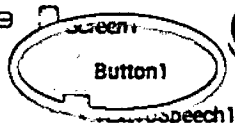
Blocks

Viewer

Built-in

- ☐ Control
- ☐ Logic
- ☐ Math
- ☒ Text
- ☐ Lists
- ☐ Colors
- ☐ Variables
- ☒ Procedures

Screen1



Any component

2

when Button1 ▾ Click

do

when Button1 ▾ GotFocus

do

when Button1 ▾ LongClick

do

when Button1 ▾ LostFocus

do

Button1 ▾ BackgroundColor ▾

set Button1 ▾ BackgroundColor ▾ to

Button1 ▾ Enabled ▾

set Button1 ▾ Enabled ▾ to

3

when Button1 ▾ Click

do



Program the TextToSpeech action

Click on the TextToSpeech drawer. Click and hold the **call TextToSpeech1.Speak** block. Drag it over to the workspace and drop it there. This is the block that will make the phone speak. Because it is inside the Button.Click, it will run when the button on your app is clicked.



MIT App Inventor 2
Beta

Project ▾ Connect ▾ Build ▾ Help ▾

My Projects Guide Report an Issue

TalkToMe

Screen1 ▾

Add Screen ...

Remove Screen

Blocks

⊞ Built-in

- ☐ Control
- ☐ Logic
- ☐ Math
- ☒ Text
- ☐ Lists
- ☐ Colors
- ☐ Variables
- ☒ Procedures

⊞ Screen1

1 ☒ Button1
TextToSpeech1

⊞ Any component

Viewer

when TextToSpeech1 AfterSpeaking
result
do

when TextToSpeech1 BeforeSpeaking
do

2 call TextToSpeech1 Speak
message

TextToSpeech1 Country

set TextToSpeech1 Country to

TextToSpeech1 Language

3 when Button1 Click
do call TextToSpeech1 Speak
message



Fill in the message socket on TextToSpeech.Speak Block

Almost done! Now you just need to tell the TextToSpeech.Speak block what to say. To do that, click on the Text drawer, drag out a **text** block and plug it into the socket labeled "message".

TalkToMe Screen1 ▾ Add Screen ... Remove Screen

Blocks

⊖ Built-in

- ☐ Control
- ☐ Logic
- ☒ **Text**
- ☐ Lists
- ☐ Colors
- ☐ Variables
- ☐ Procedures

⊖ Screen1

- ☐ Button1
- ☐ TextToSpeech1

Viewer

The screenshot shows the MIT App Inventor interface. On the left is the 'Blocks' palette, and on the right is the 'Viewer'. In the 'Text' drawer of the 'Blocks' palette, the 'Text' block is circled. In the 'Viewer', a 'Click' event block is connected to a 'TextToSpeech1.Speak' block. The 'message' socket of the 'Speak' block is circled, and an arrow points from the 'Text' block in the palette to this socket. Other blocks visible in the Viewer include 'join', 'length', 'is empty', 'compare texts', and 'trim'.

Specify what the app should say when the button is clicked

Click on the text block and type in "Congratulations! You've made your first app." (Feel free to use any phrase you like, this is just a suggestion.)

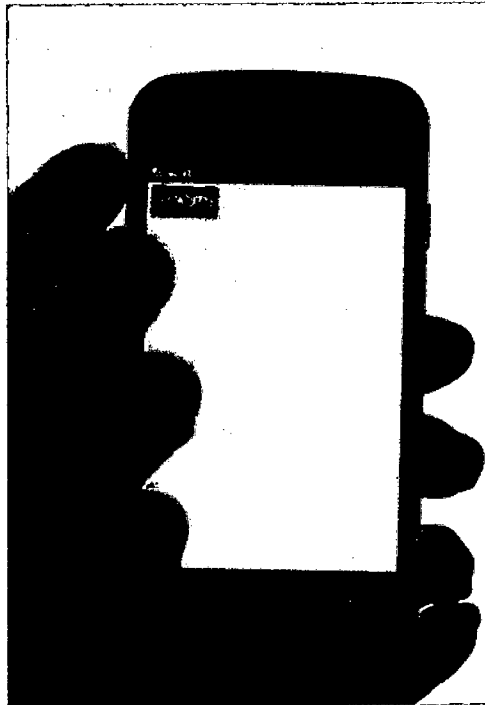
The screenshot shows a 'When Button1 Clicked' event block. Inside the event block, there is a 'call TextToSpeech1.Speak' block. The 'message' socket of the 'Speak' block is filled with the text 'Congratulations! You've made your first app.'



MIT App Inventor
appinventor.mit.edu

Now test it out!

Go to your connected device and click the button. Make sure your volume is up! You should hear the phone speak the phrase out loud. (This works even with the emulator.)



Great job!

Now move on to TalkToMe Part 2 to make the app respond to shaking and to let users put in whatever phrase they want.

LM35 Precision Centigrade Temperature Sensors

FEATURES

- Calibrated Directly in ° Celsius (Centigrade)
- Linear + 10 mV/°C Scale Factor
- 0.5°C Ensured Accuracy (at +25°C)
- Rated for Full –55°C to +150°C Range
- Suitable for Remote Applications
- Low Cost Due to Wafer-Level Trimming
- Operates from 4 to 30 V
- Less than 60-μA Current Drain
- Low Self-Heating, 0.08°C in Still Air
- Nonlinearity Only ±¼°C Typical
- Low Impedance Output, 0.1 Ω for 1 mA Load

DESCRIPTION

The LM35 series are precision integrated-circuit temperature sensors, with an output voltage linearly proportional to the Centigrade temperature. Thus the LM35 has an advantage over linear temperature sensors calibrated in ° Kelvin, as the user is not required to subtract a large constant voltage from the output to obtain convenient Centigrade scaling. The LM35 does not require any external calibration or trimming to provide typical accuracies of ±¼°C at room temperature and ±¾°C over a full –55°C to +150°C temperature range. Low cost is assured by trimming and calibration at the wafer level. The low output impedance, linear output, and precise inherent calibration of the LM35 make interfacing to readout or control circuitry especially easy. The device is used with single power supplies, or with plus and minus supplies. As the LM35 draws only 60 μA from the supply, it has very low self-heating of less than 0.1°C in still air. The LM35 is rated to operate over a –55°C to +150°C temperature range, while the LM35C is rated for a –40°C to +110°C range (–10° with improved accuracy). The LM35 series is available packaged in hermetic TO transistor packages, while the LM35C, LM35CA, and LM35D are also available in the plastic TO-92 transistor package. The LM35D is also available in an 8-lead surface-mount small-outline package and a plastic TO-220 package.

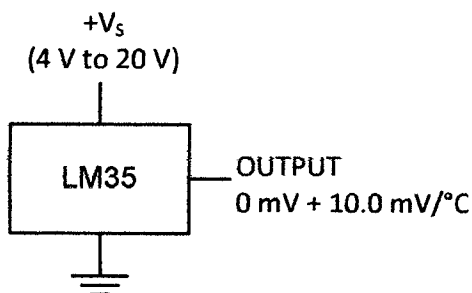
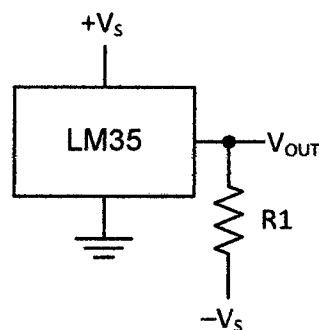


Figure 1. Basic Centigrade Temperature Sensor (+2°C to +150°C)



Choose $R_1 = -V_S / 50 \mu A$
 $V_{OUT} = 1500 \text{ mV at } 150^\circ\text{C}$
 $V_{OUT} = 250 \text{ mV at } 25^\circ\text{C}$
 $V_{OUT} = -550 \text{ mV at } -55^\circ\text{C}$

Figure 2. Full-Range Centigrade Temperature Sensor



Please be aware that an important notice concerning availability, standard warranty, and use in critical applications of Texas Instruments semiconductor products and disclaimers thereto appears at the end of this data sheet.

All trademarks are the property of their respective owners.

PRODUCTION DATA information is current as of publication date. Products conform to specifications per the terms of the Texas Instruments standard warranty. Production processing does not necessarily include testing of all parameters.

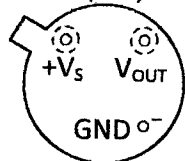
Copyright © 1999–2013, Texas Instruments Incorporated



These devices have limited built-in ESD protection. The leads should be shorted together or the device placed in conductive foam during storage or handling to prevent electrostatic damage to the MOS gates.

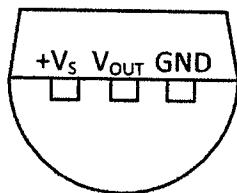
CONNECTION DIAGRAMS

**METAL CAN PACKAGE
TO (NDV)**

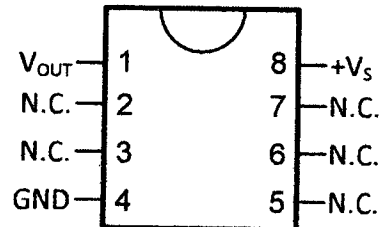


Case is connected to negative pin (GND)

**PLASTIC PACKAGE
TO-92 (LP)
BOTTOM VIEW**

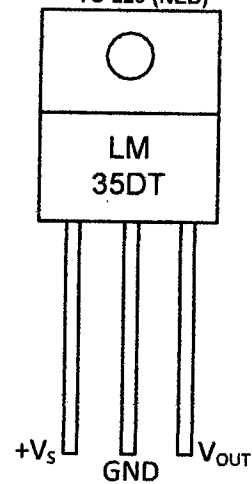


**SMALL-OUTLINE MOLDED PACKAGE
SOIC-8 (D)
TOP VIEW**



N.C. = No connection

**PLASTIC PACKAGE
TO-220 (NEB)**



Tab is connected to the negative pin (GND).

NOTE: The LM35DT pinout is different than the discontinued LM35DP

TECHNICAL DATA**MQ-2 GAS SENSOR****FEATURES**

Wide detecting scope
Stable and long life

Fast response and High sensitivity
Simple drive circuit

APPLICATION

They are used in gas leakage detecting equipments in family and industry, are suitable for detecting of LPG, i-butane, propane, methane ,alcohol, Hydrogen, smoke.

SPECIFICATIONS**A. Standard work condition**

Symbol	Parameter name	Technical condition	Remarks
V _c	Circuit voltage	5V±0.1	AC OR DC
V _H	Heating voltage	5V±0.1	ACOR DC
R _L	Load resistance	can adjust	
R _H	Heater resistance	33 Ω ±5%	Room Tem
P _H	Heating consumption	less than 800mw	

B. Environment condition

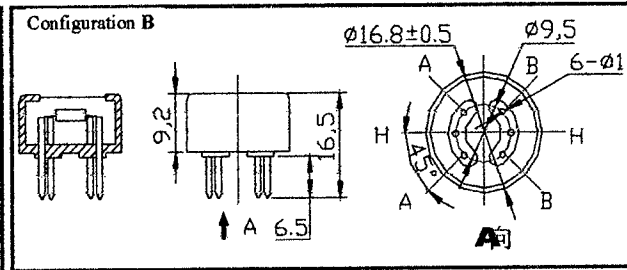
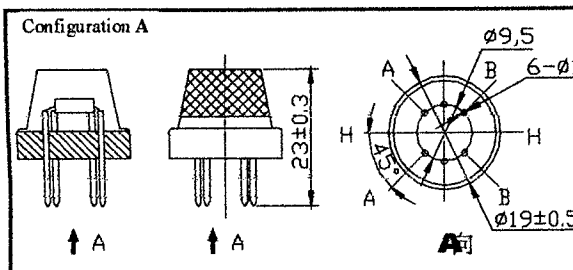
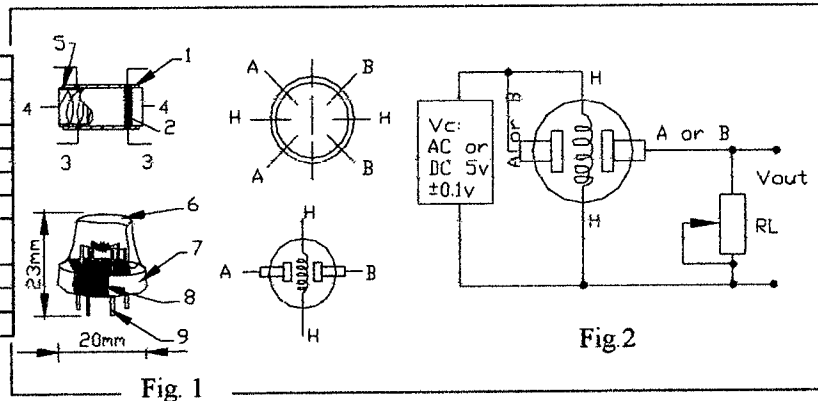
Symbol	Parameter name	Technical condition	Remarks
T _{ao}	Using Tem	-20℃-50℃	
T _{as}	Storage Tem	-20℃-70℃	
R _H	Related humidity	less than 95%Rh	
O ₂	Oxygen concentration	21%(standard condition)Oxygen concentration can affect sensitivity	minimum value is over 2%

C. Sensitivity characteristic

Symbol	Parameter name	Technical parameter	Remarks
Rs	Sensing Resistance	3K Ω-30K Ω (1000ppm iso-butane)	Detecting concentration scope: 200ppm-5000ppm LPG and propane 300ppm-5000ppm butane 5000ppm-20000ppm methane 300ppm-5000ppm H ₂ 100ppm-2000ppm Alcohol
α (3000/1000) isobutane	Concentration Slope rate	≤0.6	
Standard Detecting Condition	Temp: 20℃ ±2℃ Vc: 5V±0.1 Humidity: 65%±5% Vh: 5V±0.1		
Preheat time	Over 24 hour		

D. Structure and configuration, basic measuring circuit

Parts	Materials
1 Gas sensing layer	SnO ₂
2 Electrode	Au
3 Electrode line	Pt
4 Heater coil	Ni-Cr alloy
5 Tubular ceramic	Al ₂ O ₃
6 Anti-explosion network	Stainless steel gauze (SUS316 100-mesh)
7 Clamp ring	Copper plating Ni
8 Resin base	Bakelite
9 Tube Pin	Copper plating Ni



Structure and configuration of MQ-2 gas sensor is shown as Fig. 1 (Configuration A or B), sensor composed by micro AL₂O₃ ceramic tube, Tin Dioxide (SnO₂) sensitive layer, measuring electrode and heater are fixed into a

crust made by plastic and stainless steel net. The heater provides necessary work conditions for work of sensitive components. The enveloped MQ-2 have 6 pin ,4 of them are used to fetch signals, and other 2 are used for providing heating current.

Electric parameter measurement circuit is shown as Fig.2

E. Sensitivity characteristic curve

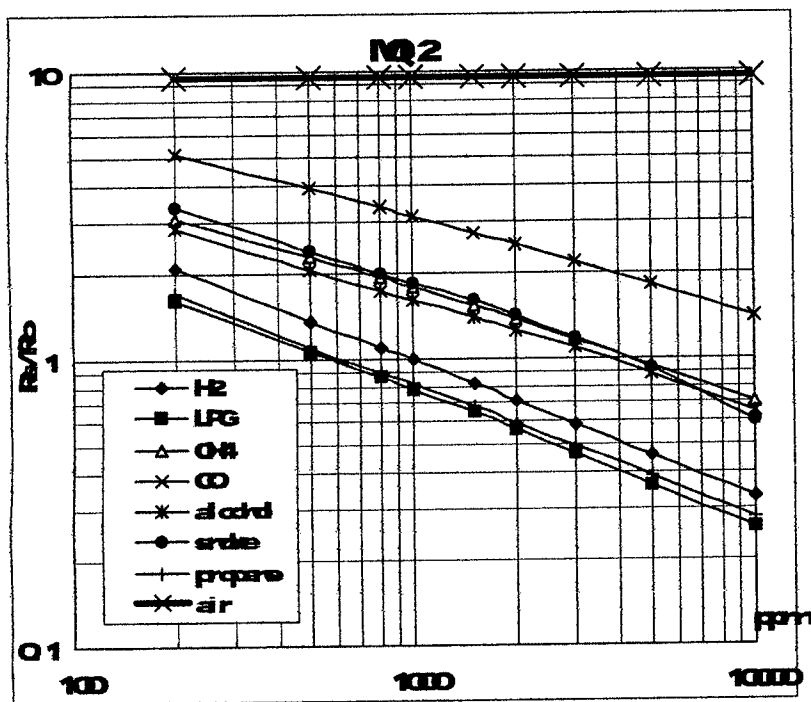


Fig.2 sensitivity characteristics of the MQ-2

Fig.3 is shows the typical sensitivity characteristics of the MQ-2 for several gases.

in their: Temp: 20°C、

Humidity: 65%、

O₂ concentration 21%

RL=5k Ω

R₀: sensor resistance at 1000ppm of H₂ in the clean air.

R_s:sensor resistance at various concentrations of gases.

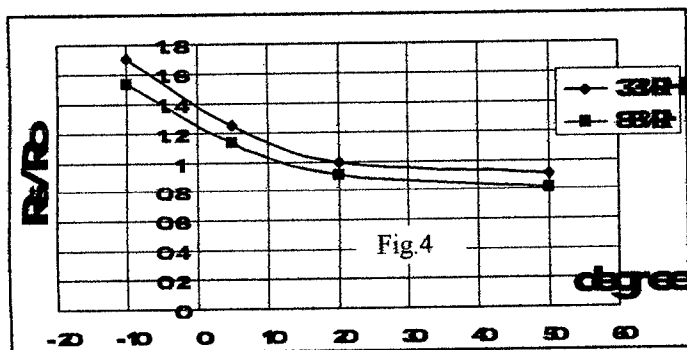


Fig.4 is shows the typical dependence of the MQ-2 on temperature and humidity.

R₀: sensor resistance at 1000ppm of H₂ in air at 33%RH and 20 degree.

R_s: sensor resistance at 1000ppm of H₂ at different temperatures and humidities.

SENSITIVITY ADJUSTMENT

Resistance value of MQ-2 is difference to various kinds and various concentration gases. So, When using this components, sensitivity adjustment is very necessary. we recommend that you calibrate the detector for 1000ppm liquified petroleum gas<LPG>, or 1000ppm iso-butane<i-C₄H₁₀> concentration in air and use value of Load resistance that(R_L) about 20 K Ω (5K Ω to 47 K Ω).

When accurately measuring, the proper alarm point for the gas detector should be determined after considering the temperature and humidity influence.

TECHNICAL DATA**MQ-3 GAS SENSOR****FEATURES**

- * High sensitivity to alcohol and small sensitivity to Benzene .
- * Fast response and High sensitivity
- * Stable and long life
- * Simple drive circuit

APPLICATION

They are suitable for alcohol checker, Breathalyser.

SPECIFICATIONS**A. Standard work condition**

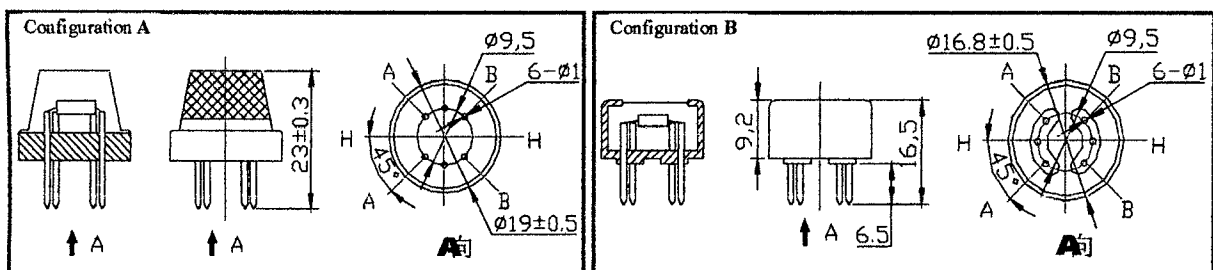
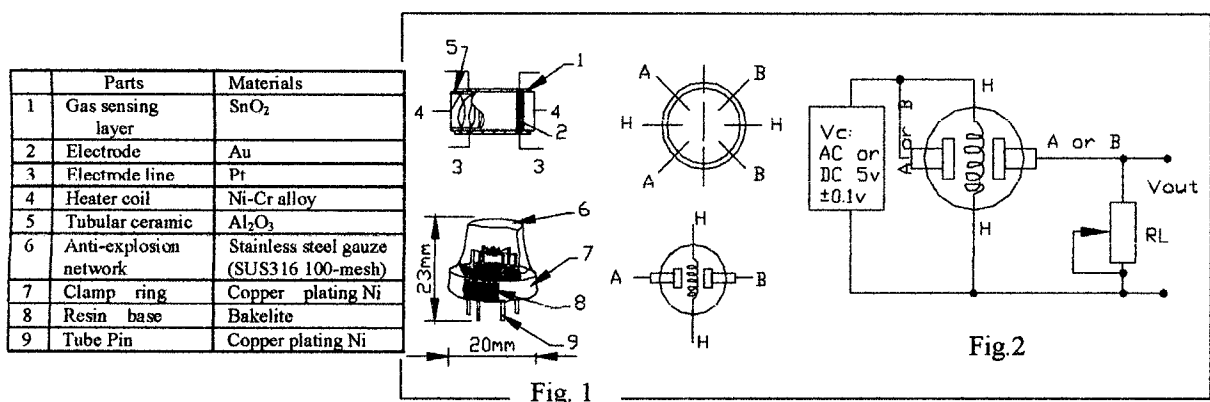
Symbol	Parameter name	Technical condition	Remarks
V_C	Circuit voltage	$5V \pm 0.1$	AC OR DC
V_H	Heating voltage	$5V \pm 0.1$	AC OR DC
R_L	Load resistance	200K Ω	
R_H	Heater resistance	$33 \Omega \pm 5\%$	Room Tem
P_H	Heating consumption	less than 750mw	

B. Environment condition

Symbol	Parameter name	Technical condition	Remarks
T_{ao}	Using Tem	$-10^\circ\text{C} - 50^\circ\text{C}$	minimum value is over 2%
T_{as}	Storage Tem	$-20^\circ\text{C} - 70^\circ\text{C}$	
R_H	Related humidity	less than 95%Rh	
O_2	Oxygen concentration	21%(standard condition) Oxygen concentration can affect sensitivity	

C. Sensitivity characteristic

Symbol	Parameter name	Technical parameter	Remarks
Rs	Sensing Resistance	1M Ω - 8 M Ω (0.4mg/L alcohol)	Detecting concentration scope: 0.05mg/L—10mg/L Alcohol
α (0.4/1 mg/L)	Concentration slope rate	≤ 0.6	
Standard detecting condition	Temp: 20℃ \pm 2℃ Humidity: 65% \pm 5%	Vc:5V \pm 0.1 Vh: 5V \pm 0.1	
Preheat time	Over 24 hour		

D. Structure and configuration, basic measuring circuit

Structure and configuration of MQ-3 gas sensor is shown as Fig. 1 (Configuration A or B), sensor composed by micro Al_2O_3 ceramic tube, Tin Dioxide (SnO_2) sensitive layer, measuring electrode and heater are fixed into a crust made by plastic and stainless steel net. The heater provides necessary work conditions for work of sensitive components. The enveloped MQ-3 have 6 pin, 4 of them are used to fetch signals, and other 2 are used for providing heating current.

Electric parameter measurement circuit is shown as Fig.2

E. Sensitivity characteristic curve

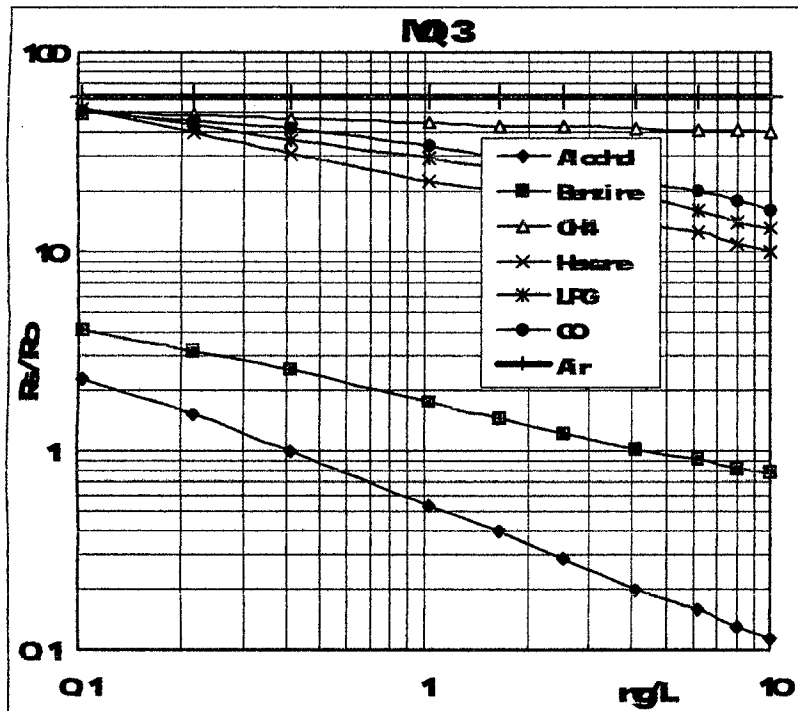


Fig.2 sensitivity characteristics of the MQ-3

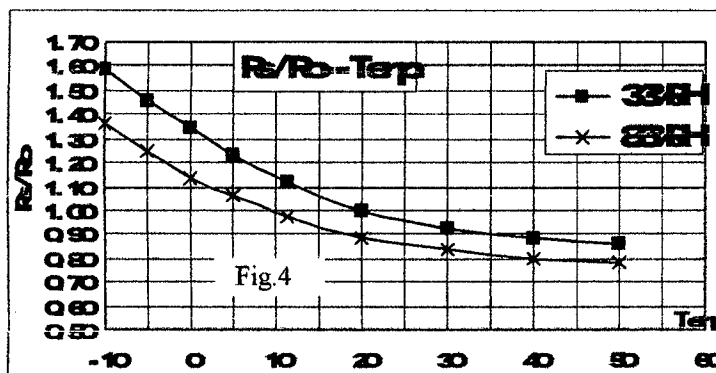


Fig.4 is shows the typical dependence of the MQ-3 on temperature and humidity.

R_0 : sensor resistance at 0.4mg/L of Alcohol in air at 33%RH and 20 °C

R_s : sensor resistance at 0.4mg/L of Alcohol at different temperatures and humidities.

SENSITIVITY ADJUSTMENT

Resistance value of MQ-3 is difference to various kinds and various concentration gases. So, When using this components, sensitivity adjustment is very necessary. we recommend that you calibrate the detector for 0.4mg/L (approximately 200ppm) of Alcohol concentration in air and use value of Load resistance that (R_L) about 200 K Ω (100K Ω to 470 K Ω).

When accurately measuring, the proper alarm point for the gas detector should be determined after considering the temperature and humidity influence.



PIC16F87XA

Data Sheet

**28/40/44-Pin Enhanced Flash
Microcontrollers**

Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as "unbreakable."

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

Information contained in this publication regarding device applications and the like is intended through suggestion only and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. No representation or warranty is given and no liability is assumed by Microchip Technology Incorporated with respect to the accuracy or use of such information, or infringement of patents or other intellectual property rights arising from such use or otherwise. Use of Microchip's products as critical components in life support systems is not authorized except with express written approval by Microchip. No licenses are conveyed, implicitly or otherwise, under any intellectual property rights.

Trademarks

The Microchip name and logo, the Microchip logo, Accuron, dsPIC, KEELoq, MPLAB, PIC, PICmicro, PICSTART, PRO MATE and PowerSmart are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

Amplab, FilterLab, microID, MXDEV, MXLAB, PICMASTER, SEEVAL and The Embedded Control Solutions Company are registered trademarks of Microchip Technology Incorporated in the U.S.A.

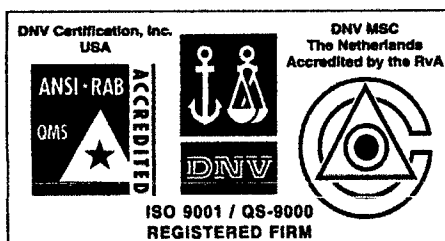
Application Maestro, dsPICDEM, dsPICDEM.net, ECAN, ECONOMONITOR, FanSense, FlexROM, fuzzyLAB, In-Circuit Serial Programming, ICSP, ICEPIC, microPort, Migratable Memory, MPASM, MPLIB, MPLINK, MPSIM, PICkit, PICDEM, PICDEM.net, PowerCal, PowerInfo, PowerMate, PowerTool, rLAB, rPIC, Select Mode, SmartSensor, SmartShunt, SmartTel and Total Endurance are trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

Serialized Quick Turn Programming (SQTP) is a service mark of Microchip Technology Incorporated in the U.S.A.

All other trademarks mentioned herein are property of their respective companies.

© 2003, Microchip Technology Incorporated, Printed in the U.S.A., All Rights Reserved.

 Printed on recycled paper.



Microchip received QS-9000 quality system certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona in July 1999 and Mountain View, California in March 2002. The Company's quality system processes and procedures are QS-9000 compliant for its PICmicro® 8-bit MCUs, KEELoq® code hopping devices, Serial EEPROMs, microperipherals, non-volatile memory and analog products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001 certified.



PIC16F87XA

28/40/44-Pin Enhanced Flash Microcontrollers

Devices Included in this Data Sheet:

- PIC16F873A
- PIC16F874A
- PIC16F876A
- PIC16F877A

High-Performance RISC CPU:

- Only 35 single-word instructions to learn
- All single-cycle instructions except for program branches, which are two-cycle
- Operating speed: DC – 20 MHz clock input
DC – 200 ns instruction cycle
- Up to 8K x 14 words of Flash Program Memory,
Up to 368 x 8 bytes of Data Memory (RAM),
Up to 256 x 8 bytes of EEPROM Data Memory
- Pinout compatible to other 28-pin or 40/44-pin
PIC16CXXX and PIC16FXXX microcontrollers

Peripheral Features:

- Timer0: 8-bit timer/counter with 8-bit prescaler
- Timer1: 16-bit timer/counter with prescaler,
can be incremented during Sleep via external
crystal/clock
- Timer2: 8-bit timer/counter with 8-bit period
register, prescaler and postscaler
- Two Capture, Compare, PWM modules
 - Capture is 16-bit, max. resolution is 12.5 ns
 - Compare is 16-bit, max. resolution is 200 ns
 - PWM max. resolution is 10-bit
- Synchronous Serial Port (SSP) with SPI™
(Master mode) and I²C™ (Master/Slave)
- Universal Synchronous Asynchronous Receiver
Transmitter (USART/SCI) with 9-bit address
detection
- Parallel Slave Port (PSP) – 8 bits wide with
external \overline{RD} , \overline{WR} and \overline{CS} controls (40/44-pin only)
- Brown-out detection circuitry for
Brown-out Reset (BOR)

Analog Features:

- 10-bit, up to 8-channel Analog-to-Digital
Converter (A/D)
- Brown-out Reset (BOR)
- Analog Comparator module with:
 - Two analog comparators
 - Programmable on-chip voltage reference
(VREF) module
 - Programmable input multiplexing from device
inputs and internal voltage reference
 - Comparator outputs are externally accessible

Special Microcontroller Features:

- 100,000 erase/write cycle Enhanced Flash
program memory typical
- 1,000,000 erase/write cycle Data EEPROM
memory typical
- Data EEPROM Retention > 40 years
- Self-reprogrammable under software control
- In-Circuit Serial Programming™ (ICSP™)
via two pins
- Single-supply 5V In-Circuit Serial Programming
- Watchdog Timer (WDT) with its own on-chip RC
oscillator for reliable operation
- Programmable code protection
- Power saving Sleep mode
- Selectable oscillator options
- In-Circuit Debug (ICD) via two pins

CMOS Technology:

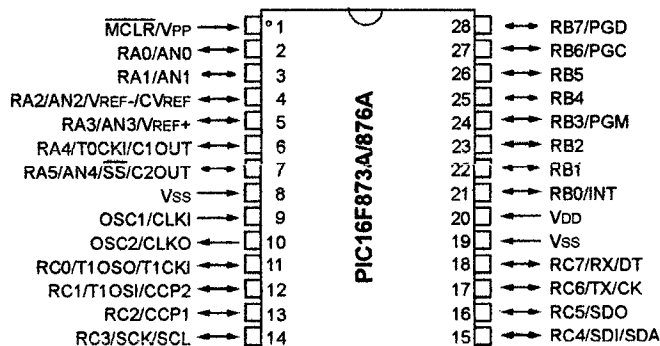
- Low-power, high-speed Flash/EEPROM
technology
- Fully static design
- Wide operating voltage range (2.0V to 5.5V)
- Commercial and Industrial temperature ranges
- Low-power consumption

Device	Program Memory		Data SRAM (Bytes)	EEPROM (Bytes)	I/O	10-bit A/D (ch)	CCP (PWM)	MSSP		USART	Timers 8/16-bit	Comparators
	Bytes	# Single Word Instructions						SPI	Master I²C			
PIC16F873A	7.2K	4096	192	128	22	5	2	Yes	Yes	Yes	2/1	2
PIC16F874A	7.2K	4096	192	128	33	8	2	Yes	Yes	Yes	2/1	2
PIC16F876A	14.3K	8192	368	256	22	5	2	Yes	Yes	Yes	2/1	2
PIC16F877A	14.3K	8192	368	256	33	8	2	Yes	Yes	Yes	2/1	2

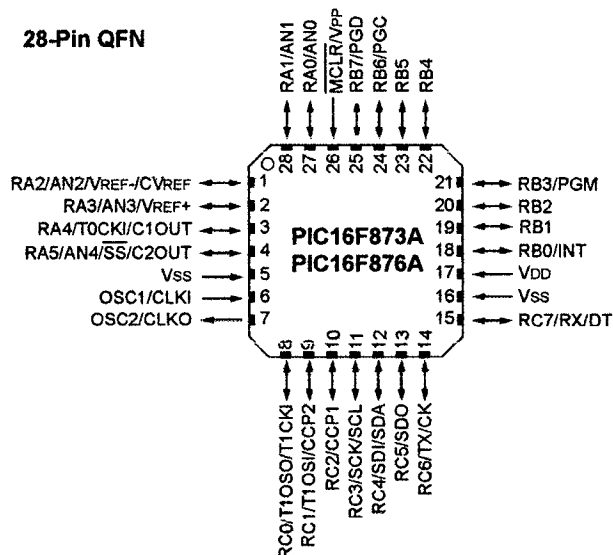
PIC16F87XA

Pin Diagrams

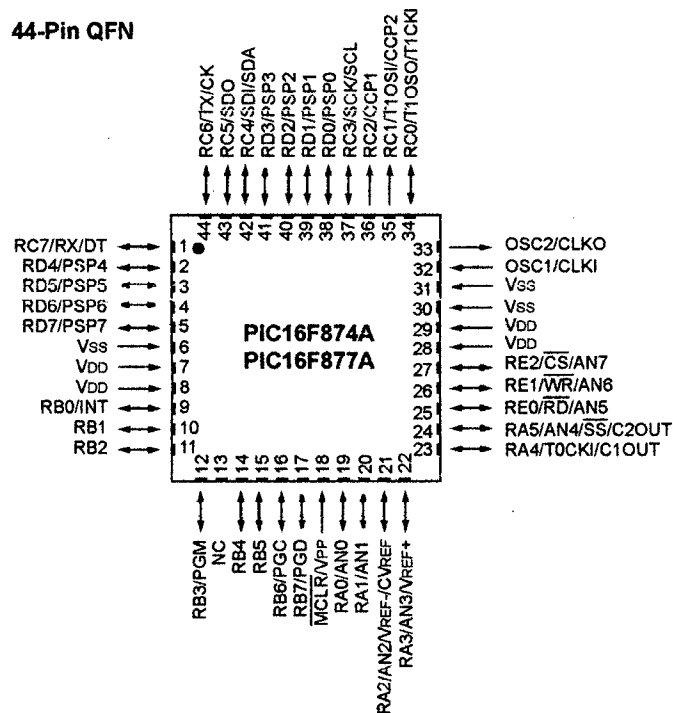
28-Pin PDIP, SOIC, SSOP



28-Pin QFN

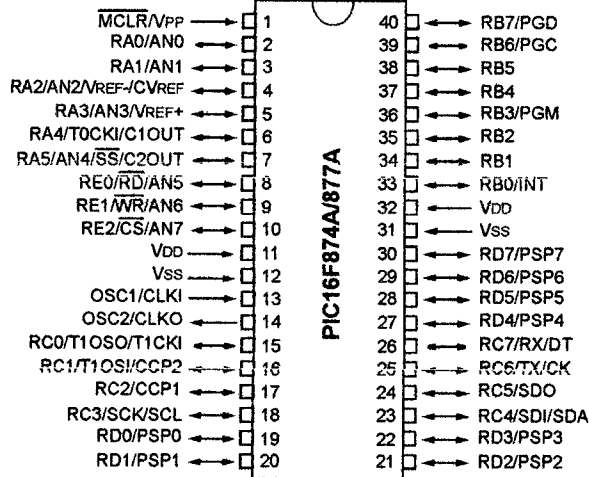


44-Pin QFN

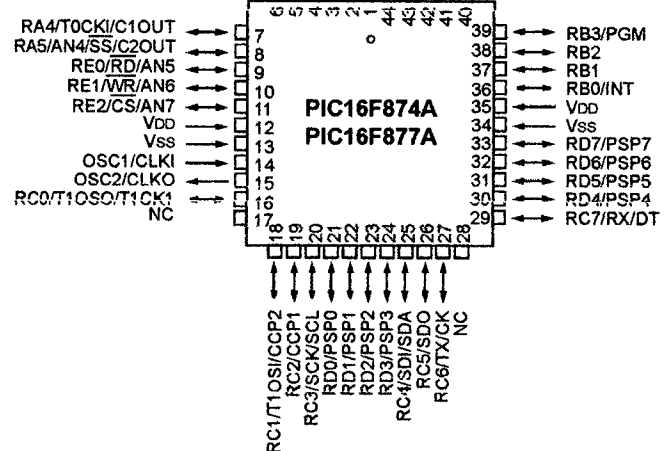


Pin Diagrams (Continued)

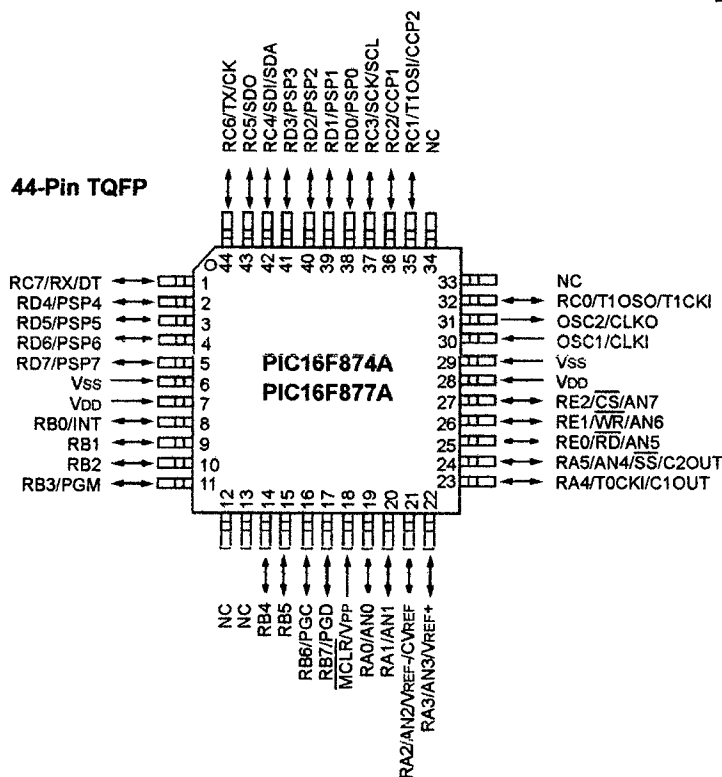
40-Pin PDIP



44-Pin PLCC



44-Pin TQFP



PIC16F87XA

Table of Contents

1.0 Device Overview.....	5
2.0 Memory Organization.....	15
3.0 Data EEPROM and Flash Program Memory.....	33
4.0 I/O Ports.....	41
5.0 Timer0 Module.....	53
6.0 Timer1 Module.....	57
7.0 Timer2 Module.....	61
8.0 Capture/Compare/PWM Modules.....	63
9.0 Master Synchronous Serial Port (MSSP) Module.....	71
10.0 Addressable Universal Synchronous Asynchronous Receiver Transmitter (USART).....	111
11.0 Analog-to-Digital Converter (A/D) Module.....	127
12.0 Comparator Module.....	135
13.0 Comparator Voltage Reference Module.....	141
14.0 Special Features of the CPU.....	143
15.0 Instruction Set Summary.....	159
16.0 Development Support.....	167
17.0 Electrical Characteristics.....	173
18.0 DC and AC Characteristics Graphs and Tables.....	197
19.0 Packaging Information.....	209
Appendix A: Revision History.....	219
Appendix B: Device Differences.....	219
Appendix C: Conversion Considerations.....	220
Index.....	221
On-Line Support.....	229
Systems Information and Upgrade Hot Line.....	229
Reader Response.....	230
PIC16F87XA Product Identification System.....	231

TO OUR VALUED CUSTOMERS

It is our intention to provide our valued customers with the best documentation possible to ensure successful use of your Microchip products. To this end, we will continue to improve our publications to better suit your needs. Our publications will be refined and enhanced as new volumes and updates are introduced.

If you have any questions or comments regarding this publication, please contact the Marketing Communications Department via E-mail at docerrors@mail.microchip.com or fax the Reader Response Form in the back of this data sheet to (480) 792-4150. We welcome your feedback.

Most Current Data Sheet

To obtain the most up-to-date version of this data sheet, please register at our Worldwide Web site at:

<http://www.microchip.com>

You can determine the version of a data sheet by examining its literature number found on the bottom outside corner of any page. The last character of the literature number is the version number, (e.g., DS30000A is version A of document DS30000).

Errata

An errata sheet, describing minor operational differences from the data sheet and recommended workarounds, may exist for current devices. As device/documentation issues become known to us, we will publish an errata sheet. The errata will specify the revision of silicon and revision of document to which it applies.

To determine if an errata sheet exists for a particular device, please check with one of the following:

- Microchip's Worldwide Web site; <http://www.microchip.com>
- Your local Microchip sales office (see last page)
- The Microchip Corporate Literature Center; U.S. FAX: (480) 792-7277

When contacting a sales office or the literature center, please specify which device, revision of silicon and data sheet (include literature number) you are using.

Customer Notification System

Register on our Web site at www.microchip.com/cn to receive the most current information on all of our products.

PIC16F87XA

1.0 DEVICE OVERVIEW

This document contains device specific information about the following devices:

- PIC16F873A
- PIC16F874A
- PIC16F876A
- PIC16F877A

PIC16F873A/876A devices are available only in 28-pin packages, while PIC16F874A/877A devices are available in 40-pin and 44-pin packages. All devices in the PIC16F87XA family share common architecture with the following differences:

- The PIC16F873A and PIC16F874A have one-half of the total on-chip memory of the PIC16F876A and PIC16F877A
- The 28-pin devices have three I/O ports, while the 40/44-pin devices have five
- The 28-pin devices have fourteen interrupts, while the 40/44-pin devices have fifteen
- The 28-pin devices have five A/D input channels, while the 40/44-pin devices have eight
- The Parallel Slave Port is implemented only on the 40/44-pin devices

The available features are summarized in Table 1-1. Block diagrams of the PIC16F873A/876A and PIC16F874A/877A devices are provided in Figure 1-1 and Figure 1-2, respectively. The pinouts for these device families are listed in Table 1-2 and Table 1-3.

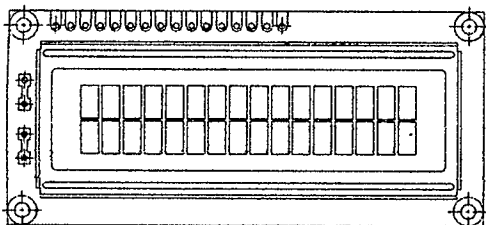
Additional information may be found in the PICmicro® Mid-Range Reference Manual (DS33023), which may be obtained from your local Microchip Sales Representative or downloaded from the Microchip web site. The Reference Manual should be considered a complementary document to this data sheet and is highly recommended reading for a better understanding of the device architecture and operation of the peripheral modules.

TABLE 1-1: PIC16F87XA DEVICE FEATURES

Key Features	PIC16F873A	PIC16F874A	PIC16F876A	PIC16F877A
Operating Frequency	DC – 20 MHz	DC – 20 MHz	DC – 20 MHz	DC – 20 MHz
Resets (and Delays)	POR, BOR (PWRT, OST)	POR, BOR (PWRT, OST)	POR, BOR (PWRT, OST)	POR, BOR (PWRT, OST)
Flash Program Memory (14-bit words)	4K	4K	8K	8K
Data Memory (bytes)	192	192	368	368
EEPROM Data Memory (bytes)	128	128	256	256
Interrupts	14	15	14	15
I/O Ports	Ports A, B, C	Ports A, B, C, D, E	Ports A, B, C	Ports A, B, C, D, E
Timers	3	3	3	3
Capture/Compare/PWM modules	2	2	2	2
Serial Communications	MSSP, USART	MSSP, USART	MSSP, USART	MSSP, USART
Parallel Communications	—	PSP	—	PSP
10-bit Analog-to-Digital Module	5 input channels	8 input channels	5 input channels	8 input channels
Analog Comparators	2	2	2	2
Instruction Set	35 Instructions	35 Instructions	35 Instructions	35 Instructions
Packages	28-pin PDIP 28-pin SOIC 28-pin SSOP 28-pin QFN	40-pin PDIP 44-pin PLCC 44-pin TQFP 44-pin QFN	28-pin PDIP 28-pin SOIC 28-pin SSOP 28-pin QFN	40-pin PDIP 44-pin PLCC 44-pin TQFP 44-pin QFN



16 x 2 Character LCD



FEATURES

- 5 x 8 dots with cursor
- Built-in controller (KS 0066 or Equivalent)
- + 5V power supply (Also available for + 3V)
- 1/16 duty cycle
- B/L to be driven by pin 1, pin 2 or pin 15, pin 16 or A.K (LED)
- N.V. optional for + 3V power supply

MECHANICAL DATA

ITEM	STANDARD VALUE	UNIT
Module Dimension	80.0 x 36.0	mm
Viewing Area	66.0 x 16.0	mm
Dot Size	0.56 x 0.66	mm
Character Size	2.96 x 5.56	mm

ABSOLUTE MAXIMUM RATING

ITEM	SYMBOL	STANDARD VALUE			UNIT
		MIN.	TYP.	MAX.	
Power Supply	VDD-VSS	- 0.3	—	7.0	V
Input Voltage	VI	- 0.3	—	VDD	V

NOTE: VSS = 0 Volt, VDD = 5.0 Volt

ELECTRICAL SPECIFICATIONS

ITEM	SYMBOL	CONDITION	STANDARD VALUE			UNIT
			MIN.	TYP.	MAX.	
Input Voltage	VDD	VDD = + 5V	4.7	5.0	5.3	V
		VDD = + 3V	2.7	3.0	5.3	V
Supply Current	IDD	VDD = 5V	—	1.2	3.0	mA
Recommended LC Driving Voltage for Normal Temp. Version Module	VDD - V0	- 20 °C	—	—	—	V
		0°C	4.2	4.8	5.1	
		25°C	3.8	4.2	4.6	
		50°C	3.6	4.0	4.4	
		70°C	—	—	—	
LED Forward Voltage	VF	25°C	—	4.2	4.6	V
LED Forward Current	IF	25°C Array	—	130	260	mA
		Edge	—	20	40	
EL Power Supply Current	IEL	Vel = 110VAC:400Hz	—	—	5.0	mA

DISPLAY CHARACTER ADDRESS CODE:

Display Position	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
DD RAM Address	00	01														0F
DD RAM Address	40	41														4F

LCD-016M002B

Vishay

16 x 2 Character LCD



PIN NUMBER	SYMBOL	FUNCTION
1	Vss	GND
2	Vdd	+ 3V or + 5V
3	Vo	Contrast Adjustment
4	RS	H/L Register Select Signal
5	R/W	H/L Read/Write Signal
6	E	H → L Enable Signal
7	DB0	H/L Data Bus Line
8	DB1	H/L Data Bus Line
9	DB2	H/L Data Bus Line
10	DB3	H/L Data Bus Line
11	DB4	H/L Data Bus Line
12	DB5	H/L Data Bus Line
13	DB6	H/L Data Bus Line
14	DB7	H/L Data Bus Line
15	A/Vee	+ 4.2V for LED/Negative Voltage Output
16	K	Power Supply for B/L (OV)

DIMENSIONS in millimeters

